NIXIES! are back in town

May 2011

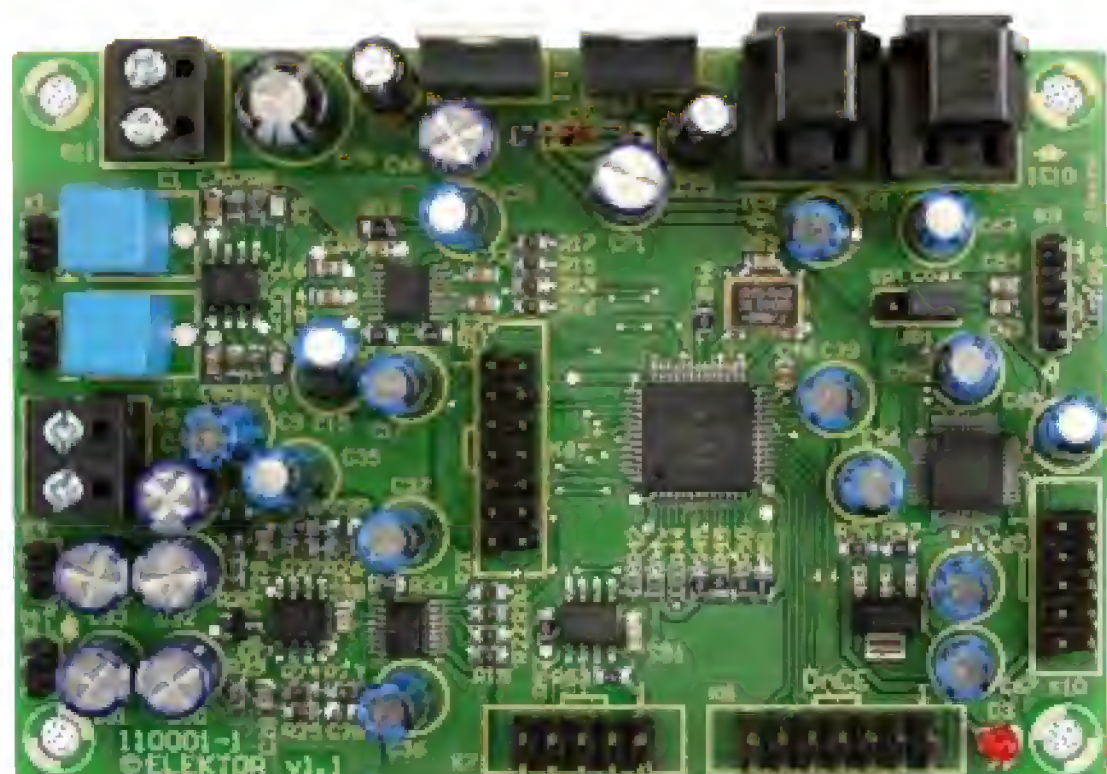AUS$ 14.50 - NZ$ 17.50 - SAR 102.95 - NOK 99   £ 4.80

+ Design professional front panels for free

elektor
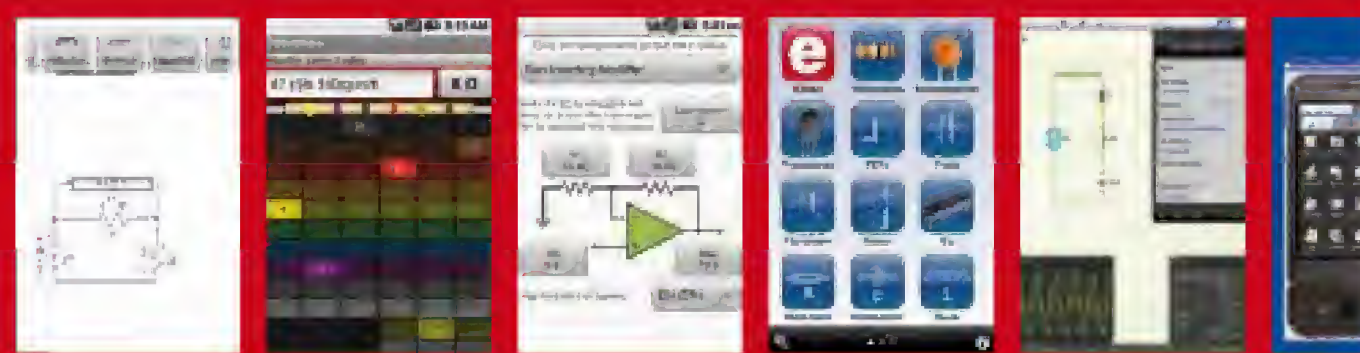
www.elektor.com

Elektor Proton Robot

A versatile platform for learning and experimenting

Audio DSP Course

Fast forward in audio signal processing

How useful are electronics apps for tablets and smartphones?

+ 1-Channel DMX512 Light Dimmer

+ Microphone Conferencing System

LPC1200

Cortex-M0 – A simple choice

- Up to 128 KB Flash with 512 Byte page erase and 8 KB SRAM

- Industry's first ROM-based divide library for Cortex™-M0, offering fast and deterministic execution for 32-bit division

- Unique configurable peripherals suited to energy efficient applications

- Supported by LPCXpresso tool chain for easy development and migration

www.nxp.com/cortex-m0

Digital Audio

Before the compact disc (CD) appeared on the market, sound editing involved nothing but analog signals. In essence a tiny alternating voltage from a microphone or a turntable stylus got sent through a number of gain stages before being applied to a tape recorder head, a disc cutter or a loudspeaker drive unit. These days audio editing and recording is considerably more complex, although consumers fail to realize it (excluding of course Elektor readers!). Today even the cheapest MP3 player is sure to contain a microprocessor or a DSP that performs advanced signal processing to enable memory resident data to arrive on your eardrums in analogue form.

For sure, today's DSPs are very powerful cores suitable for all kinds (audio) applications, but at the same time they're highly complex beasts to deal with. Consequently it's not at all easy to design a DSP-based circuit and associated software if you're not at home in this area. The DSP course launched in this edition we hope spells the end of fears and ignorance surrounding DSPs. Although we intend to make practical aspects prevail, inevitably some interaction with theory is required considering that everything in this course revolves around real circuits with a DSP inside. Along the course several example projects will be discussed, all of which can be replicated at home or in the lab using a universal DSP board developed by the author in close collaboration with Elektor Labs. Having digested the course in a few months time you should be ready to start developing custom applications for the DSP board. In this edition we kick off the course with some theory you really can't get around before the programming and soldering can commence. So bite the bullet and join our DSP course, it's sure to be fun and educational.

Enjoy reading this edition,
Jan Buiting, Editor

CONTENTS

elektor | international media bv

Elektor International Media provides a multimedia and interactive platform for everyone interested in electronics. From professionals passionate about their work to enthusiasts with professional ambitions. From beginner to diehard, from student to lecturer. Information, education, inspiration and entertainment. Analogue and digital; practical and theoretical; software and hardware.

ANALOGUE • DIGITAL
MICROCONTROLLERS & EMBEDDED
AUDIO • TEST & MEASUREMENT

Second generation high voltage battery stack monitor

Linear Technology announces the LTC®6803, a second generation high voltage battery monitor for hybrid/electric vehicle (HEVs), electric vehicles (EVs) and other high voltage, high performance battery systems, The LTC6803 is a complete battery measuring IC that includes a 12-bit ADC, a precision voltage reference, a high voltage input multiplexer and a serial interface. Each LTC6803 can measure up to 12 individual battery cells in series. The device's proprietary design enables multiple LTC6803s to be stacked in series without optocouplers or isolators, permitting precision voltage monitoring of every cell in long strings of series-connected batteries. The LTC6803 follows the road-proven LTC6802, introduced back in 2008 with the same functionality and pinout, plus a number of significant performance enhancements.

The maximum total measurement error of the LTC6803 is guaranteed to be less than 0.25% from −40°C to 125°C. The LTC6803 offers an extended cell measurement range from -300mV to 5V, enabling the LTC6803 to monitor a wide range of battery chemistries, as well as supercapacitors, Each cell is monitored for undervoltage and overvoltage conditions, and an associated MOSFET is available to discharge overcharged cells, Added functionality is provided by an onboard 5V regulator, temperature sensor, GPIO lines and thermistor inputs.

Erik Soule, Vice President, Signal Conditioning Products, stated, "Now over two years in the field with our proven battery stack monitor family, we are pleased to offer our customers this enhanced product, the LTC6803. Automakers and other customers worldwide have embraced Linear's robust battery stack monitors, and we are confident that this new addition to the family will provide excellent performance for the next generation of HEVs and EVs.

For long-term battery pack storage, the current consumed by the integrated BMS can potentially unbalance the cells, The LTC6803 addresses this concern with a standby mode that draws less than 12uA, Furthermore, the power input of the LTC6803 is isolated from the stack, allowing the LTC6803 to draw current from an independent source, When powering from this input, the current draw on the pack is reduced to less than 1µA.

The LTC6803 is designed to surpass the environmental, reliability and safety demands of automotive and industrial applications. The device is fully specified for operation from −40°C to 125°C, It has been engineered for ISO 26262 (ASIL) compliant systems and a full set of self-tests ensure that there are no latent fault conditions, To meet this standard, the LTC6803 includes a redundant voltage reference, extensive logic test circuitry, open wire detection capability and a watchdog timer for fail-safe designs, The LTC6803 is designed to withstand up to 75 V, providing more than 20% of overvoltage margin for a full string of 12 cells. The 1MHz serial interface includes packet error checking and is designed to operate in the presence of large amounts of noise and transients.

The LTC6803 is offered in a small 8mm x 12mm surface mount device, Priced at $9.95 each in 1,000-piece quantities, samples, demonstration boards and the data sheet are now available at www.linear.com/product/LTC6803-1. The LTC6803 is now available in production quantities.

(110048-VII)   www.linear.com/product/LTC6803-1

Step-down converter with linear bypass powers multimode PAs in mobile handsets

Maxim Integrated Products introduces the MAX8989, a step-down converter designed to power PA (power amplifier) modules in mobile phones. The first DC-DC converter in the industry to support power management for multiple communication protocols (e.g., LTE, WCDMA, GSM, and
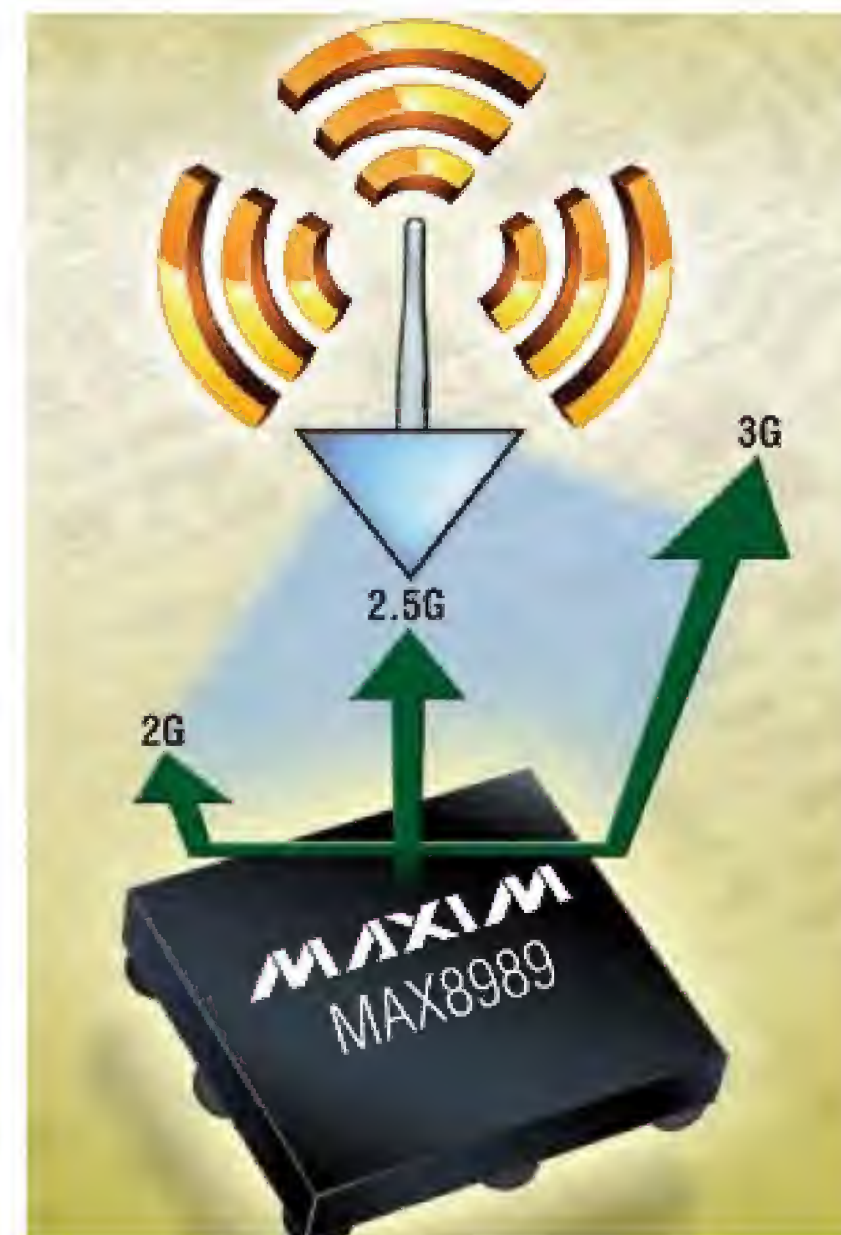
EDGE), the MAX8989 is ideal for converged (multimode) PAs. This solution is utilized in Infineon's proven SMARTi(TM) UE2 RF engine platform, which delivers unparalleled flexibility and efficiency to multistandard cellular phone systems.

The MAX8989 integrates a high-efficiency switching regulator with a parallel bypass LDO to guarantee maximum efficiency under all conditions. This approach enables designers to minimize the amount of power dissipated within the PA by regulating the PA's output stage supply voltage. By controlling the PA supply voltage to meet transmission-power requirements, designers can extend 2G talk time by up to 25% and 3G talk time by up to 10%.

The integrated LDO serves to smooth the transition between step-down regulation and operation in dropout, allowing for a seamless transition to higher power levels. When in dropout, the LDO becomes a low-impedance (85 milliohm) bypass FET to ensure minimal power loss during periods of peak current demand. Additionally, the switching regulator is capable of a 100% duty cycle to further eliminate losses.

The MAX8989 supports up to 2.5 A, making it an excellent choice for powering GSM/EDGE PAs. It is fully specified over the −40 degrees Celsius to +85 degrees Celsius extended temperature range, and offered in a compact (1.6mm × 1.6mm, 0.69mm max height), 9-bump wafer-level package (WLP). Samples and evaluation kits are available upon request.

(110048-IX)   www.maxim-ic.com/MAX8989

'Go anywhere' powerful computing capability

The key differences between BVM's new Mobile PC family and the ubiquitous laptop, tablet and Smartphone units lies in the fact that Mobile PC is a full industrial computer, that, by integrating several different technologies, is able to accept inputs from a range of sources, store and manipulate the data and act as a communications hub. In addition to its PC capabilities, it provides, as standard, 3.5G, Wi-Fi and Bluetooth; GPS is also embedded. Its key strengths are the provision of wire-free internet access combined with powerful computing, enabling it to be deployed across a wide range of scenarios: the possible applications are virtually endless.

- Providing computing capability in a Portacabin at the start of a green field development.
- Capturing video images, time and GPS location data on buses as a defence against fraudulent personal injury compensation claims.
- Controlling motorway signage displays.
- Interfacing with radio-controlled fire detection systems on construction sites.
- Storing images from IP-based security cameras.
- Acting as a server for location-specific offers in shopping centres, sending a special-offer text message to people's phones as they pick up the Wi-FI signal.
- Machine-to-Machine communications: embedded or stand-alone mounting to control and manage functionality, capture and communicate data from sensors and provide interoperability with other machines.

All these applications, which demonstrate the wide scope of potential uses for the BVM Mobile PC units, are currently either in use in the field or under discussion with national and international customers.

Designed, developed and manufactured in the UK, as standard, there are various combinations of processor, memory and I/O options available for Mobile PC. Whatever combinations are chosen, a key feature of the family is that all versions are provided with extensive communications capabilities. They support multiple 3 and 3.5G WAN protocols: HSUPA, 3G offering up to 5.7 Mbit/s; HSDPS 3.5G up to 14.4 Mbit/s and UMTS 3.5G with a theoretical 45 Mbit/s download bandwidth. Wi-Fi, conforming to the 802.11b/g/n specifications, offering 11, 54 and 600 Mbit/s raw data rates and Bluetooth short-range radio connectivity are built in. Embedded GPS capability significantly extends the functionality of the units into mobile applications.

The entry level OB100 is based on an Intel Atom D510 dual core processor, supports up to 2GB of Ram and has embedded video and high definition audio. 4 x USB2.0 ports and support for wired Ethernet and wireless capability with both active and passive antennas are provided. The top of the range OB200 is based on the 64-bit Core i7/i5/i3 processors, up to 8GB of RAM, 2 serial ports, 10 x USB2.0 ports, a LCD inverter port and a SMBUS port for communication with the PSU and other system elements.

A further capability from BVM Mobile is the iKarPC, a self-contained unit specifically designed for bus and truck fleet operators. It interfaces with the vehicle's CANbus, supports three major protocols and monitors and stores more than 79 different parameters such as vehicle speed, engine loading, rpm and fuel level for remote interrogation by the central fleet management system, which can also locate and track the vehicle fleet, plan routes, progress jobs and generate reports. Again, comprehensive communications facilities are provided with GPS, Bluetooth, Wi-Fi and 2.5 and 3.5G WAN capability. For long distance hauliers, in addition to the business functionality of the device, DVM-T compatible Mobile Entertainment capability provides multi-channel TV and radio through the unit for when they are parked overnight.

The BVM Mobile PC family and the iKarPC are significant advances in mobile computing, offering simple and cost-effective solutions to the previously intractable problems associated with providing significant computing power to remote and mobile locations.

www.bvmltd.co.uk   (110270-II)

Cypress: Charger Armor avoids false coordinates and phantom touches common with inexpensive chargers

Cypress Semiconductor Corp. unveiled a breakthrough feature for its TrueTouch™ touchscreen controllers that enables touchscreens in handsets, cameras, GPS systems and other mobile systems to function seamlessly while connected to noisy chargers. Low-cost, third-party and after-market battery chargers emit large levels of common-mode noise that adversely affects touchscreen performance, often resulting in inaccurate touch readings or phantom touches. Cypress's new Charger Armor™ feature allows TrueTouch touchscreens to operate normally in the presence of extreme AC noise.

The new feature, which is available in Cypress's CY8CTMA3XX TrueTouch family, is unique in the industry, and solves a common problem that mobile device manufacturers are struggling to overcome. Low cost chargers lack critical noise suppression components, and can generate tens of volts (peak to peak) across the frequency spectrum of 1 Hz to 1 MHz. These voltage spikes can go directly into the touch panel during the presence of touch, severely impacting touch performance. As such, many of the mobile phone vendors have worked together to create EN 62684 and EN301489, standards which regulate the noise spectrum for battery chargers. While Cypress's TrueTouch technology
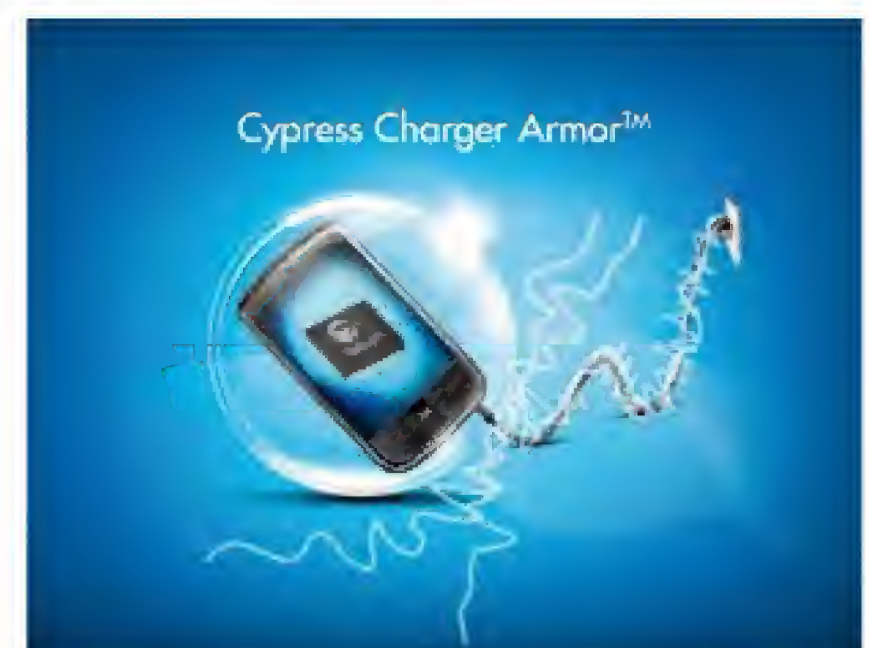
alone can suppress noise at the levels of these standards, Charger Armor allows Cypress to go beyond the standards to deliver truly revolutionary charger noise immunity to the market, enabling mobile devices to operate with the lowest-cost chargers.

(110048-X)   www.cypress.com/go/TrueTouch

Anritsu: new handheld spectrum analyzer type MS2711E

Anritsu Company recently introduced the MS2711E Spectrum Master, the latest handheld spectrum analyzer designed for technicians, installers, field RF engineers, and contractors responsible for monitoring the growing number of interfering signals, as well as assessing signal quality on a wide range of increasingly complex signals. Easy to use, integrated and high performing, the MS2711E offers users the performance and features necessary to meet their testing challenges.

The MS2711E comes equipped with a touch-screen interface, easy to use short-cut icons and intuitive menus, all designed to minimize training time and enhance the user experience. The latest Spectrum Master model also has built-in system level measurements, such as
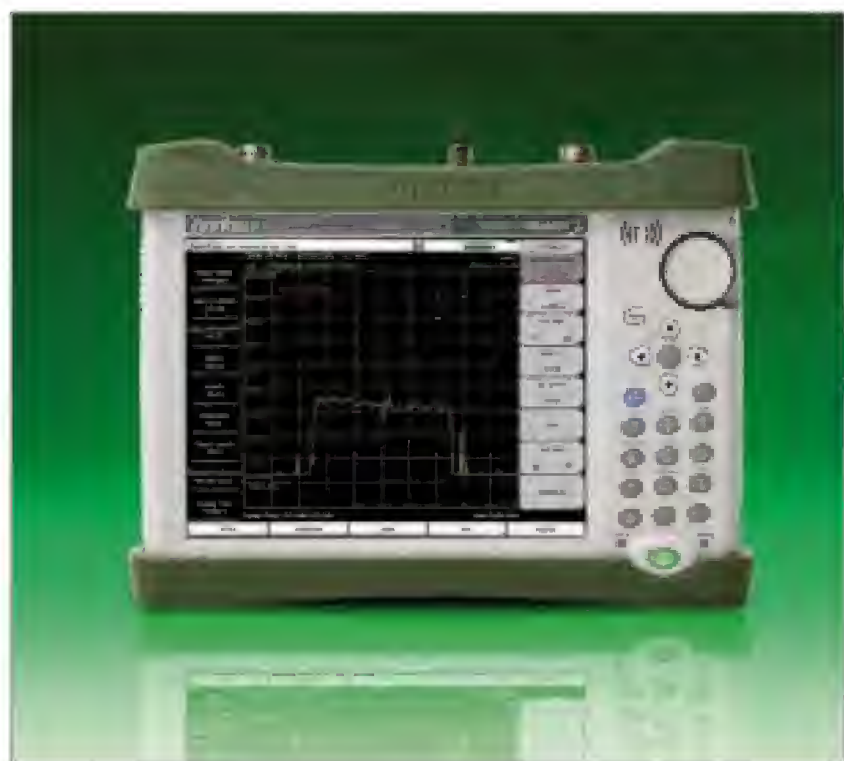
Occupied Bandwidth, Adjacent Channel Power Ratio, Emission Mask, and Field Strength measurements. Additionally, the MS2711E can be configured with various options, such as an interference analyzer, channel scanner, power meter and GPS receiver, to create an integrated solution that eliminates the need to carry multiple instruments.

IAR: LPC1 Kickstart kit

The new IAR KickStart Kit for LPC1 includes a development board with an ARM Cortex-M0 based LPC1227 microcontroller, peripherals and connectors, an IAR J-Link Lite debug probe providing SWD debugging, software development tools and board support packages for various RTOSes.

This is believed to be the world's first commercial starter kit for the ARM Cortex-M0-based LPC1227 microcontroller. The strong partnership and tight cooperation between NXP and IAR Systems during the development project has led to IAR Systems latest starter kit being the first to the market.

Included in the kit is a code size limited version of IAR Embedded Workbench, a set of development tools for building and debugging embedded system applications using assembler, C and C++. It provides a completely integrated development environment that includes a project manager, editor, build tools and the C-SPY debugger.

IAR KickStart Kit for LPC1227 is priced at £129 / $169. It can be bought online at www.iar.com/eshop.

www.iar.com   (110270-III)

Rugged and lightweight, the MS2711E is designed for field use. Weighing less that 3.5 kg, the handheld spectrum analyzer provides a bright 8.4-inch touch-screen display. Operating temperature is –10° to 55° C with a warm-up temperature of less than five minutes, easily surpassing available competitive products. Battery life is greater than three hours.

High-end performance has been maintained in the compact design. The MS2711E provides the best accuracy in its class, so field technicians can confidently conduct spectrum monitoring, broadcast proofing and RF/microwave measurements. Covering the 100 kHz to 3 GHz frequency range, the MS2711E has Displayed Average Noise Level (DANL) of >–142 dBm in 100 Hz RBW, dynamic range of >85 dB to measure small signals in the presence of larger ones, and frequency accuracy of <± 50 ppb with GPS. Master Software Tools (MST) is provided with each MS2711E, as well as all Spectrum Master instruments. MST provides a simple and easy method to manage, archive, analyze, and print measurement reports, and keep the analyzer current with the latest firmware. MST also allows users to transfer data collected on their MS2711E to a PC for easy post-processing analysis. Anritsu also offers certified training courses in basic spectrum analysis, interference analysis and other options and features.

(110270-I)   www.anritsu.com

GRM Tools Version 3 bundles

GRM Tools proudly announces that they have partnered with DontCrac[k] to announce the awaited release of the GRM3 Tools version 3 bundles.

Not only the famous "CLASSIC" and "SPECTRAL TRANSFORM" bundles have been totally re-written from scratch and are now available in the RTAS – AU – VST and Standalone formats for Mac & PC, but the folks at GRM Tools have designed an incredible set of new plug-ins called "EVOLUTION".

The GRM Tools "EVOLUTION" bundle is perhaps the most original and unexpected piece of audio software made in the past decade. EVOLUTION allows powerful and unexpected spectral and harmonic audio transformations such as morphing, evolving textures, original "VOCODER" type effects, time shifted filters, sliding timbers, noising, etc. EVOLUTION can also be used as very subtle audio treatment effects in delicate processing situations. Describing what "EVOLUTION" really does is pretty tough! We therefore suggest you download the free demo (no iLok or authorization required) and experiment for yourself.

EVOLUTION might simply change the way Sound Designers, Engineers, Musicians or Composers will process audio in the future. GRM Tools therefore opened an official

GRM Tools forum board at DontCrac[k] for users to post their comments, bug reports and experiences using these new "unconventional" audio tools.

From Friday, March 25 to Sunday, April 17, 2011, EVOLUTION can be purchased as part of the GRM Tools Complete Collection at introductory price of only $489 (MSRP $1150). GRM Tools are actually sold and distributed exclusively by DontCrack.com via Instant Download.

(110270-IV)   www.dontcrack.com/news/

CC sized feature packed system on module

Anders Electronics, market leader in the field of user interface platforms, has announced the availability of the new CompuLab CM-T3517, a miniature system-on-module based on the Texas Instruments' Sitara AM3517/05 system-on-chip, featuring the advanced Cortex-A8 ARM CPU, which powers devices from high-end feature phones to netbooks, DTVs, printers and automotive-infotainment systems.

Ideally suited for high-visibility, space-constrained industrial and retail products like shelf-edge advertising, kiosks, dispensing machines, ticketing machines or elevator controllers — the CM-T3517 provides a rich set of multimedia, connectivity and storage features on a credit-card size system-on-module.

Leveraging the power and flexibility of the TI Sitara AM3517/05 system-on-chip, which is based on the advanced Cortex-A8 ARM CPU — a proven processor known for its high-end performance — the CM-T3517 enables embedded systems to provide the usability and connectivity that users expect, while still being extremely developer-friendly.

The CM-T3517 is offered with either a TI Sitara AM3517 or Sitara AM3505 with Cortex-A8 core (up to 600 MHz), up to 256 MB DDR2, up to 512 MB flash disk including file system protection, a touchscreen controller supporting 4-wire resistive panels, and an onboard power management chip with single 3.3v input or Lithium-ion Polymer battery.

The multimedia-ready system features PowerVR SGX GPU providing 2D/3D graphics acceleration with OpenGL-ES and OpenVG support, dual Output Graphics (RGB and TV Out), a graphics controller supporting STN and TFT panels with 1400 x 1050 max resolution, a camera interface port, and audio codec stereo line-out and microphone. For connectivity, the CM-T3517 offers a 100 Mbps Ethernet port, integral WiFi 802.11b/g, Bluetooth interface, MMC/SD/SDIO support including SDHC up to 32GB, 4 serial ports, I2C, SPI, GPIOs, and a CAN bus. Further, in addition to an on-the-go (OTG) host/slave USB2 high-speed port, the CM-T3517 offers an additional four USB2 host ports.

From a software point of view, the CM-T3517 was specially designed to serve as a building block in embedded applications, and includes all the peripherals required to run operating systems such as Linux and Windows CE.

(110270-V)   www.anders.co.uk

Market's smallest 3-axis analogue gyroscope

STMicroelectronics has extended its motion-sensor portfolio with the market's smallest 3-axis analog-output gyroscope. Housed inside an ultra-small $4x4x1mm^3$ package, ST's newest gyroscope combines superior performance and reliability with smart power management and design flexibility, advancing adoption of high-precision gesture control and intuitive user interfaces in mobile phones, tablets, game controllers and other consumer devices.
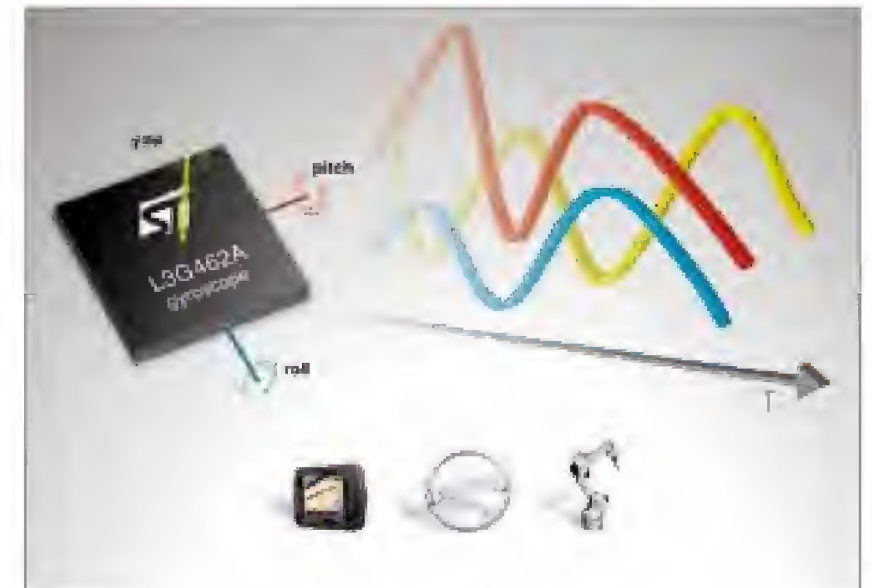
All of ST's MEMS gyroscopes employ an industry-unique concept of a single sensing structure for motion measurement along all three orthogonal axes. This innovative design approach eliminates all interference between the axes, significantly increasing accuracy and reliability in a wide range of consumer and industrial applications. The company has introduced more than 40 gyroscopes over past two years.

ST's L3G462A gyroscope provides accurate angular-rate detection with a separate output for each of the three axes. The full scale of ±625 dps accurately measures gestures and movements of different types and speeds, enabling applications from navigation to motion-activated user interface and gaming. The analogue interface ensures very fast response with low latency and offers flexibility in external filtering capabilities and fine-tuning of key parameters to the individual application requirements.

The gyroscope's extremely compact design eliminates the layout and placement issues common with existing bulkier solutions, without penalty in performance or reliability. The unparalleled stability over temperature (±0.04 dps/°C) and sensitivity (±0.017%/°C) significantly improves measurement accuracy and the low noise level (0.017 dps/√Hz) guarantees

high precision in movement and gesture recognition.

ST's newest 3-axis analog-output gyroscope can operate with any supply voltage in the range of 2.4 V to 3.6 V and the power-down and sleep-to-wake modes contribute to the reduction in power consumption of the entire system.

Volume production is scheduled for the end of Q3 2011.

(110270-VI)   www.st.com/mems

60 V step-down LED driver drives three 100mA LED strings LEDs with 10,000:1 True Color PWM dimming

Linear Technology announces the LT3597, a 60 V, 1 MHz, step-down DC/DC converter designed to operate as a three channel, constant current LED driver. Each channel of the LT3597 contains a constant current sink LED driver with a dedicated adaptive-output buck converter. This design offers maximum efficiency for applications, such as RGB displays, which require different output voltages for each LED string. From a 48 V input, the LT3597 can drive up to three strings of LEDs, each with up to ten 100 mA LEDs in series, delivering efficiencies of over 90%. The LT3597's multichannel capability makes it ideal for backlighting LED billboards as well as industrial, automotive
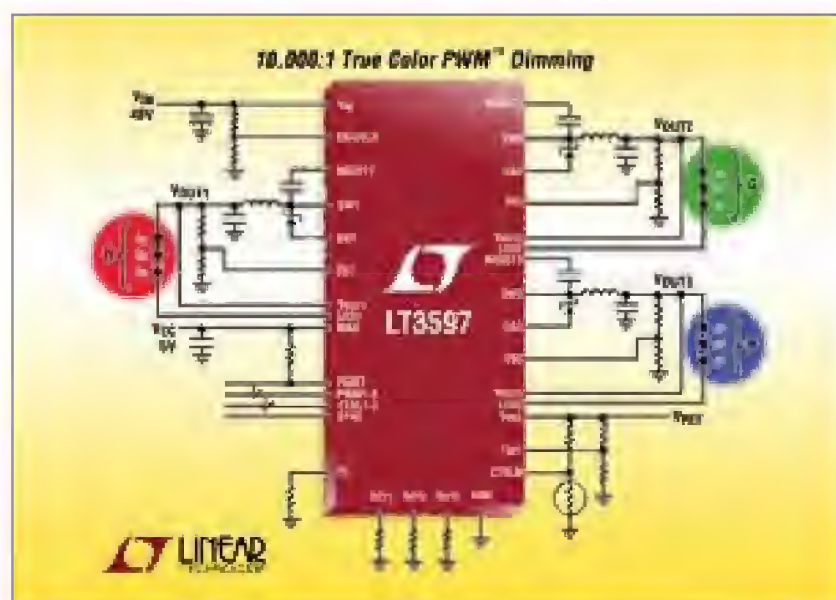
and medical displays. Similarly, its input voltage range of 6 V to 60 V makes it well suited for industrial, automotive and medical applications. Each LED string can be independently programmed to deliver up to 100mA of LED current with ± 2% current matching between strings. Furthermore, each channel can be independently dimmed with True Color PWM™ dimming, offering dimming ratios as high as 10,000:1. A

New LED lighting microsite for design engineers

Farnell has launched an innovative microsite dedicated to LED lighting technology. The new Farnell site provides engineers with a comprehensive LED lighting destination, offering product and solution-based guidance for applications such as street lighting, MR16's, solar power signals and RGB colour mixing, as well as access to a wide range of design and collaboration tools via element14, its exclusive eCommunity site for design engineers.

In addition to application notes, white papers, training resources, design tools, and videos, the new LED site enables engineers to view and purchase from a best-in-class product portfolio available for same day shipping.

Engineers have one stop access to a wide range of

LED Lighting product including LEDs, Power & Thermal Management, Interconnect, Optics and integrated/modular LED Lighting solutions. The offering consists of LED Lighting product from top brands including Cree, Avago, Bridgelux, Ledil, National Semiconductor, Linear Technology, Texas Instruments, TE Connectivity, Molex, Fisher and Bergquist.

Engineers can quickly link from the microsite to the lighting group at Farnell's element14 engineering community to collaborate with industry peers and experts. element14's lighting group offers quick access to additional technical information, lighting insight from its "Ask an Expert" panel of professionals and lively discussions with engineers from around the globe.

(110270-VII)   www.farnell.com/lighting

programmable 200 kHz to 1MHz fixed frequency operation and current-mode architecture ensures stable operation over a wide range of supply and output voltages, minimizing the size of external components. Additionally, the switching frequency is synchronizable to an external clock. The LT3597's thermally enhanced 5mm x 8mm QFN package provides a highly compact solution footprint for LED lighting applications

The LT3597 uses three independent 400 mA, 60 V internal switches to provide step-down constant current sources. The

internal buck converters use an adaptive feedback loop to regulate the output voltage slightly higher than the required LED voltage to ensure maximum efficiency. If any of the LED strings experience an open or short-circuit, the LT3597 will continue to regulate the other strings and signal the FAULT pin. Additional features include LED current derating based on junction temperature and/or LED temperature and LED disconnect in shutdown.

The LT3597EUHG is available in a 39-lead 5mm x 8mm QFN package.

(110270-XI)   www.linear.com/product/LT3597

Development Tools on the Go

How useful are electronics apps for tablets and smartphones?

A wide and rich variety of software tools is available for electronics development. These engineering aids are no longer confined to the PC: smartphones and tablet computers already have plenty of apps available for them too. So, we decided to find out what programs are available and how usable they are.

By Christopher Rausch
(Elektor Germany Editorial)

PC-based software for designing and simulating electronic circuits is now a common weapon in the engineer's arsenal. Readers will no doubt have heard of products such as Proteus and PSpice. Although these programs are capable of delivering very accurate simulation results [1], they are nevertheless often very expensive and require a PC. However, an internet search will reveal electronics applications not just for PCs, but also for smartphones and tablet computers; these are usually much cheaper and in some cases

are even offered for free. The computing power now available in mobile devices is very high, and the devices are almost exclusively operated using a touchscreen. We decided to find out what the apps are capable of and what features they offer the user. What opportunities (and what problems) does the touchscreen interface bring, and how restrictive are the relatively small displays?

To answer these questions, we took a closer look at two apps for smartphones and two for the Apple iPad. In the case of the smartphones we decided to try the free Android operating system. This OS

is very widely used and gives software developers more freedom than they have when developing for the iPhone, for example. Whereas Apple developers cannot freely distribute their programs to run on real hardware without obtaining prior approval from the manufacturer, Android developers face no such hurdles. Indeed, Android developers are not even forced to use the official Android application market. For our tests we used the Windows Android emulator (see text box) from the official Android SDK (software development kit) [2].

An overview of some other electronics applications, untested by us, appears in the

Figure 1. A voltage divider in Droid Tesla.

Figure 2. Colour-code decoder in Electronica.

inset towards the end of this article. The selection of smartphone applications is based on a list produced by Android expert Stefan Schwark.

Development tools in your pocket

The palette of electronics applications available for Android smartphones ranges from digital reference works and component value calculators to software for drawing and simulating electronic circuits. Because of the relatively small display, smartphone applications make much greater demands on the user interface in terms of understandability and ease-of-use. On the other hand, the applications make the ideal vademecum for the engineer on the move.

The first application that we will look at more closely is called Droid Tesla [3]. This software (**Figure 1**) is a circuit simulator. The user interface controls (which can be hidden) allow you to construct your circuit and then start the simulation. The simulator can currently cope with devices including resistors, capacitors, inductors and complex impedances as well as voltmeters and ammeters. Voltage and current sources are available to power the circuit. Voltage and current are specified with real and imaginary components, which is relatively unusual. Unfortunately the simulator does not provide a graphical display of its results: the values can only be read from the virtual voltmeters and ammeters. The way that various elements can be hidden on the display is convenient and saves space, though it can often be fiddly to connect components exactly as you want. The relatively small screen area available compared to a PC's monitor also makes it difficult to get a good overview of a large circuit.

The majority of electronics applications for Android smartphones are calculators and reference information for frequently-used components and circuit modules. The standard repertoire includes resistor colour-code decoders, calculators for Ohm's law, potential dividers, resonant circuits and other basic elements of

electronic designs. The second application in our test belongs to this category and is called 'Electronica' [4]. It includes a range of calculators, a colour-code decoder (**Figure 2**) and information about IC and connector pinouts. In terms of the number of different functions offered this program is one of the smaller ones of its kind, but it is easy to use and quickly leads to the required results. The individual tools can be selected using a category menu and are clearly laid out.

Circuit design on the iPad

The iPad is a tablet computer, like a smartphone with a somewhat larger (9.7 inch) display. It is almost exclusively operated using its touchscreen. A wide range of electronics applications is also available for this device. The spectrum ranges from calculator applications and component databases to circuit simulators. Again we meet the question of how successful the touchscreen is as an intuitive interface for the application, and how

suitable the iPad software is for practical use. For our test we selected two circuit simulators for the iPad. We managed to construct small circuits with the help of the software; we also managed to call up pre-loaded circuits and simulate them, with the results being displayed graphically.

The first application we will look at is Software Circuit Lab HD [5] (**Figure 3**). As almost invariably with the iPad, the touchscreen is used to enter schematics. First it is necessary to tap the points between which the component is to be connected. Then the desired component is selected from a menu and inserted into the schematic. Compared to a drag-and-drop approach, this method takes some getting used to. The components available include passives such as resistors, capacitors and inductors, as well as semiconductor devices such as diodes, operational amplifiers and bipolar transistors. The component library is thus more than adequate for small analogue circuits. DC and AC voltage and
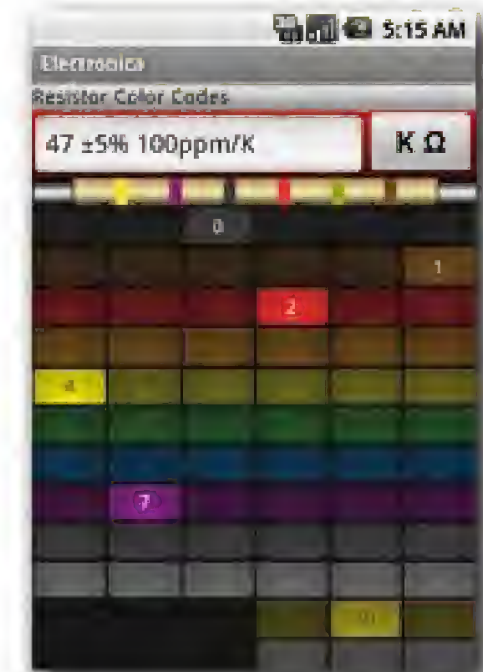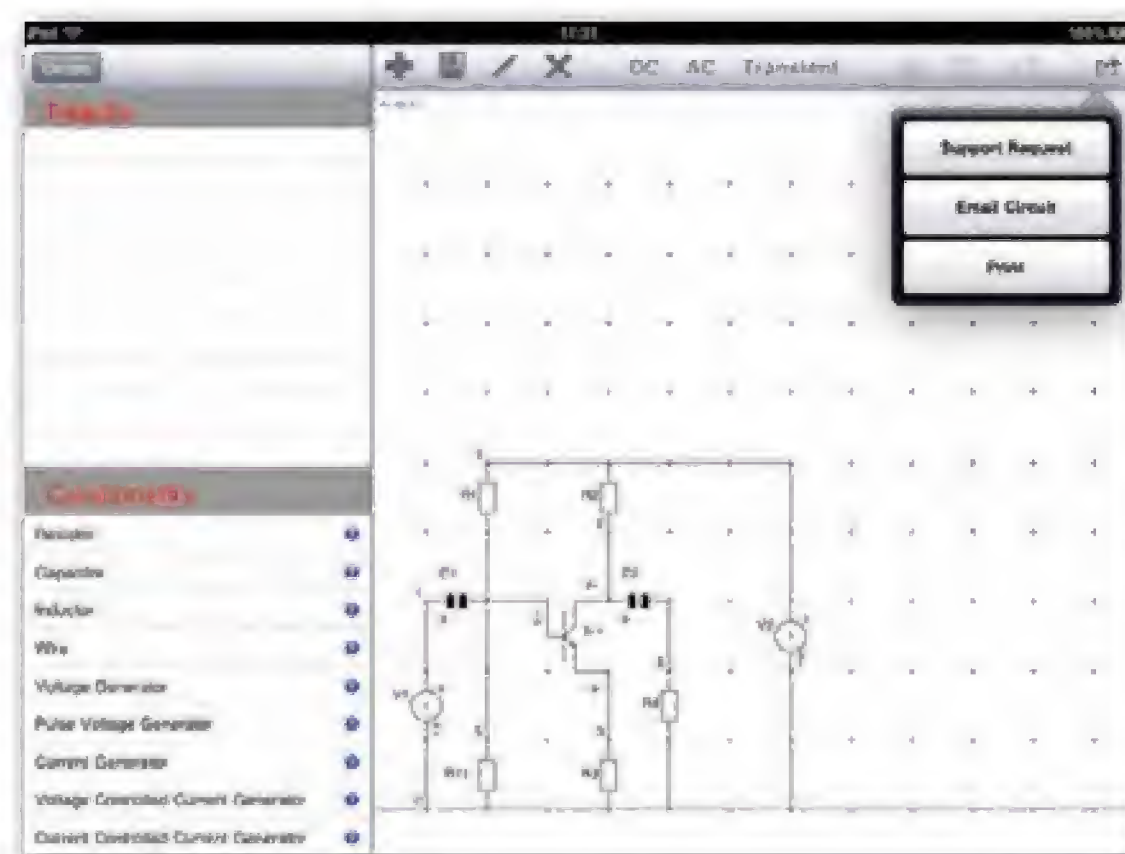
Figure 3. Transistor circuit on the iPad.

More Android and iPad applications

DCircuit Lab HD: This is the companion application to Circuit Lab HD for digital components. According to the maker, combinatorial and sequential circuits can be analysed either step by step or in one go.

(http://sites.google.com/site/lurvill/dcircuitlabhd)

Elektor Electronic Toolbox: This tool is a large collection of electronic aids, including many calculators, converters, IC datasheets, circuit diagrams and pinouts for connectors.

(www.elektor.com/news/electronic-toolbox-for-the-iphone.1412874.lynkx)

Ohm's Calculator: This is a simple application for performing calculations based on Ohm's law. A special user interface simplifies entering the desired values.

(http://smartchef.org/rantekinc/)

Scientific Electronics Circuit Calculator for iPad: This is a collection of calculators and formulae for the working electronics engineer. Results are automatically recomputed whenever a component value is changed, so that results are obtained quickly.

(http://www.syclabs.com/CircuitCalciPad.html)

Elec Ref: This is a collection of calculator tools for cables, transformers and motors, along with information about various types of cable.

(http://www.trunnion.info/elecref.html)

Electrical Wiring Pro: This is a collection of information and tools all relating to cables and cabling. A complete feature list can be found on the maker's website.

(http://www.intineo.com/electrical_wiring_pro.html)

Audio Test Tone Generator: This application includes a signal generator for sine, triangle, square and sawtooth waves as well as a range of different types of noise, with both graphical and audio output.

(http://www.digitalantics.co.uk/android/audio-test-tone-generator)

ElectroDroid: ElectroDroid includes a range of calculator tools and information (pinouts of various interface connectors) for electronics engineers.

(http://demisoft.altervista.org/_siti_interni/electrodroid/)

current sources are also available, both controllable and fixed. Component values can be entered using the iPad's on-screen keyboard. When the circuit is complete, the user can start the simulation and display the various currents and voltages around the circuit either numerically or graphically. DC, AC and transient (evolution of currents and voltages over a specified time period) simulation are available. When the software presents results graphically, several curves can be overlaid on one graph. Sadly the graphs are presented without scales, which makes interpreting the results difficult and imprecise.

Simulation as a game

Like the application described above, iCircuit [6] is a circuit simulator. However,

it gives the user the opportunity to modify the circuit while the simulation is running and observe the effect of these modifications on the live plots produced by the program. Schematics are constructed using a drag-and-drop approach: the components are dragged from the menu to the right place in the schematic using a finger, and wired to one another in a similar fashion. Component values can be entered either by using the on-screen keyboard or by using a slider (**Figure 4**). Changing a component value does not force the simulation to stop, which means that you can interact with the circuit almost like a toy, giving the software an educational feel. A wide range of components is available including standard passives as well as switches, semiconductors including
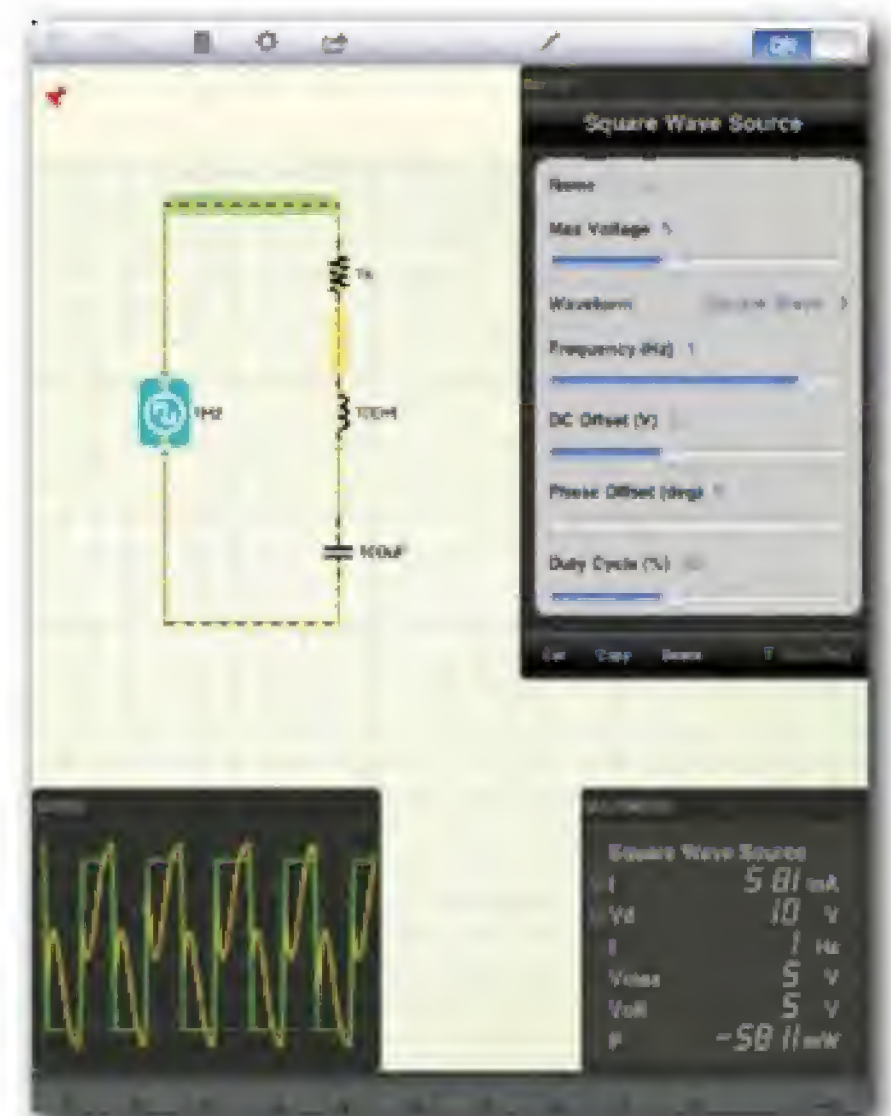
Figure 4. Simulating a circuit using iCircuit.

diodes, bipolar transistors and field-effect transistors, A/D and D/A converters and simple logic devices. Visualisation of the

Internet Links

[1] www.elektor.com/100359

[2] http://developer.android.com/guide/developing/tools/emulator.html

[3] https://market.android.com/details?id=org.vlada.droidtesla

[4] https://market.android.com/details?id=com.hacsoft.electronica

[5] http://sites.google.com/site/lurvill/circuitlabhd

[6] http://icircuitapp.com/

[7] http://developer.android.com/sdk/index.html

simulator results is via virtual oscilloscopes. These displays can be hidden and revealed as required. It is also possible to display current magnitude and direction as well as charge distribution on the wires in the schematic.

Conclusion

Developing and simulating larger circuits on smartphones is still rather difficult. The main reason for this is the relatively small size of the display on these mobile devices compared to a PC's monitor, which makes it very easy to lose sight of the overall structure of a larger circuit. The functionality of the applications is also lacking in some respects, although it is worth bearing in mind that in many cases the software is free. The smartphone is better suited to calculator and reference tools. The display is large enough for this kind of application, and it is possible to display all the necessary content in a convenient and comprehensible way. The portability of the device is also a significant plus, making it a handy tool even when out and about.

The iPad applications we looked at are suitable for constructing and simulating small circuits, but do not offer the range of functions that a professional electronics designer would find necessary. This applies both to the range of components that can be used and to the facilities for analysing the simulator output. But again, the application only costs a fraction of what one might pay for a professional piece of PC-based software. The touchscreen is an intuitive interface, but sometimes lacks the precision needed to draw circuits and to wire up components with confidence. For school pupils, students and hobbyists, however, iCircuit is a good choice. The live simulation feature and the ability to visualise current flows within the circuit makes the software a good basis for experimenting and learning.

(110230)

About the author

Christopher Rausch is a master's student in Technology and Communications, specialising in electronic engineering, at RWTH Aachen University. He is currently working as an intern with the Elektor Germany editorial team.

Android Emulator

The emulator forms part of the Android software development kit [7] and is able to simulate an Android smartphone on a PC. This is an excellent way for software developers taking their first steps in Android device programming to try out their programs quickly and easily without needing actual hardware. The user can install new applications on the system either by downloading them using a pre-installed browser, or by using a console command and an Android package (APK file) that has been previously-downloaded or built. To take the latter route, the user must (under Windows) open a DOS command window (command.exe) in addition to the emulator, and then, using the familiar DOS commands (such as 'cd <path>') switch to the directory from which the Emulator was started. Installation is then carried out using the command 'adb install <file path>' entered directly at the command window. The full path to the package file is required as a parameter, including the '.apk' suffix. When installation is complete, the new program will be found in the applications menu of the emulated smartphone.

The Nixie Tube

The design and control of Nixie tubes

By Jens Boos (Germany)

There is no doubt Nixie tubes are once again trendy. No other display technology has quite the same character. The fact that we are unlikely ever to see these devices in consumer products (unless production restarts) makes their use in a one-off design special. If you are planning to add the Nixie coolness factor to your next home brew design we give details of their operating principle, power supplies and practical advice on driving the tubes. To inspire you a collection of fascinating reader's projects has been included.

To describe a Nixie tube as an electronic device for displaying numbers or characters is a reasonably good description of its basic function. It does not however give the full picture; a bit like describing a log fire as a type of room heater. Although Nixies are essentially cold-cathode devices many people find their orange glow very appealing, evoking nostalgia and memories of a bygone era. Back in the 1950s to the 1970s before the advent of LEDs, Nixie tubes displays were standard fare on frequency counters, bench timers and most other test equipment, Elektor even featured a project using them [1].

In more recent times we have responded to the revival of interest by publishing the 'Sputnik Clock' in January 2007 [2] and the 'Nixie Tube Thermometer' in January 2011 [3]. In this article we shed some light on their operating principles and give some historical background of the device.

A little history ...

During the course of the 20th century with the rise of digital electronics it became more important for equipment to deal with numeric values and to display values as numbers rather than the using the less precise analogue meter or crude indicator lamp. At the beginning of the 1950s the Nixie tube was developed to meet this need. It can display different types of symbols, is free of mechanical wear and consumes very little power. Interestingly a patent was granted for a Nixie-like indicator device back in the 1930s [4] but was never commercially exploited at the time.

There is a certain amount of confusion over which company was the first to produce a functioning 'Nixie-like' display tube. The typical Nixie tube familiar to collectors today was first introduced in 1955 by the Burroughs Corporation who one year later registered it as a trademark [5]. The name Nixie was originally only used internally in Burroughs to stand for 'Numeric Indicator

Figure 1. Construction of a typical Nixie tube.

eXperimental no. 1' (**Figure 2**). One year earlier (1954) the company National Union had introduced its own design named the Inditron but this device remains something of a historical curiosity. The tube did not have an anode so required more complex driver electronics and it quickly lost ground to the Nixie when it was introduced.

In the following years the Nixie gained acceptance throughout the world and led to the production of the tubes in the UK, Germany, France, Poland, Russia, Japan and China, sometimes under licence to Burroughs.

The last Nixie tubes to be built rolled off the production line in Russia in the early 1990s. The company Richardson Electronics Ltd in the US reputedly still have all the necessary machines to begin production should the need arise [6].

The complete range of Nixie tubes produced is impressively large. The author has amassed a good selection of tubes over the years some of which are shown in the photo on the first page. A few of these examples are very rare and quite precious [7].

The structure and function

The Nixie tube consists of a sealed glass envelope filled with the noble gas neon or a mixture of neon and argon. Later models also included mercury vapour. Each of the display numbers has been punched from a very thin metal sheet. These are fixed in the glass envelope and wired to individual pins on the tube's base. Each number forms a cathode connection (**Figure 1**) and can be independently controlled. The anode typically consists of a fine wire mesh surrounding the numbers.

A small proportion of the neon-argon gas filling is already ionised due to the influence of the ever-present cosmic radiation. When a DC voltage is applied between the anode (plus) and cathode (minus) the electric field generated accelerates the available free charge
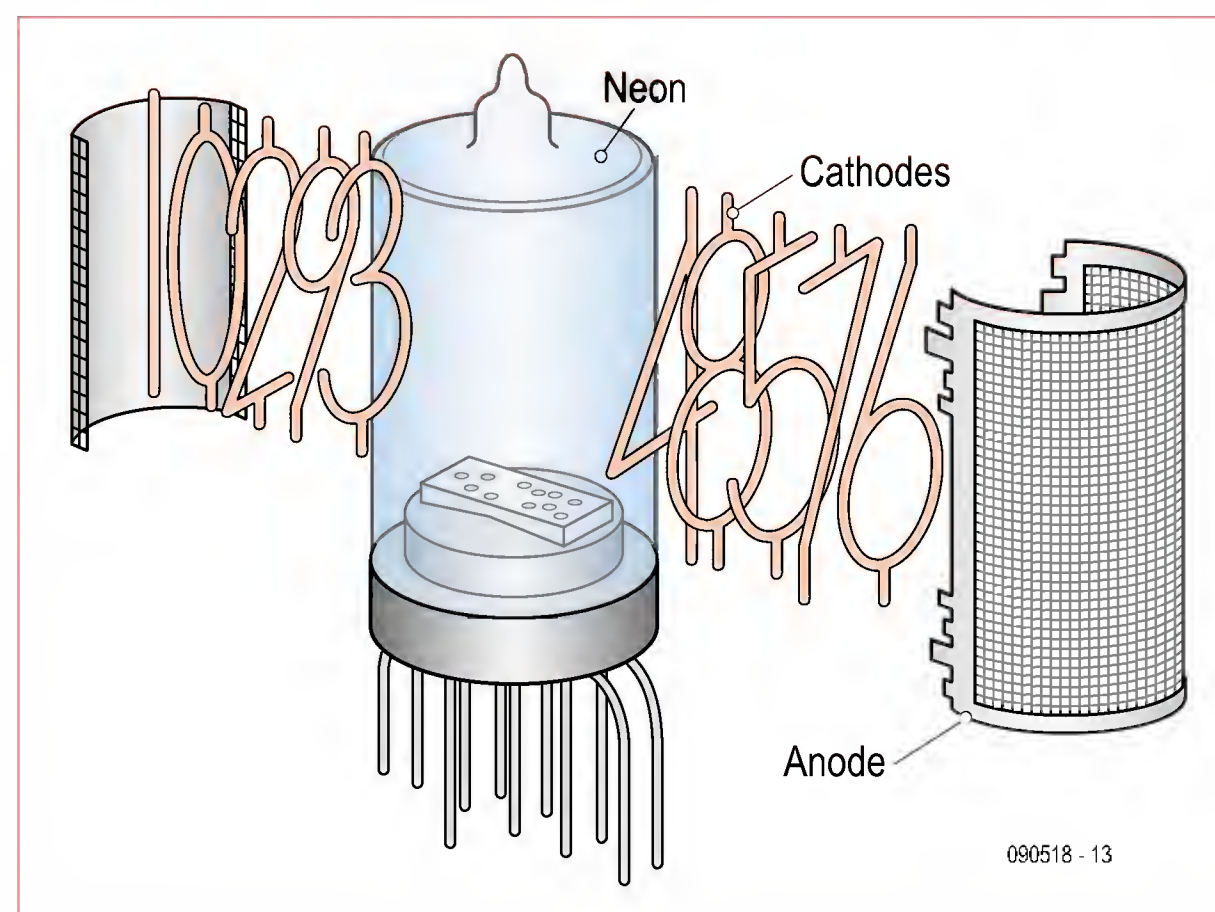
Figure 2. Advertisement for a Nixie (ca 1955).

carriers (electrons and gas ions) in the glass envelope. When this DC potential exceeds a certain value the speed of the particles (particularly electrons) is sufficient to ionise the neutral gas molecules. A charge-carrier avalanche occurs, causing significant current to flow through the tube. This 'strike voltage' is typically in the range 120 to 150 V.

Figure 3. The Burroughs HB-106 along with the GI-10 from National Union.

A surplus of positive charge carriers (positive space charge region) builds up around the cathode while these gas ions have a much higher mass than the electrons. At this point the voltage gradient and therefore the field strength is increased so that the external voltage can now be reduced to some extent while still maintaining conduction. The lowest voltage necessary to sustain conduction is called the maintaining voltage; and is typically 5 to 10 V below the strike voltage.

When the Nixie is operating at a voltage somewhere between the strike and extinguish voltages (i.e. at the *maintaining voltage*), electrons form the majority of the flow through the tube. Shortly after leaving the cathode their speed is sufficient to excite emission from the noble gas (electrons in the outer shell of the noble gas atoms are moved to a higher energy level and then release a photon as they fall back to a lower energy level). On close inspection it can be seen that the illumination in the tube does not actually occur on the surface of the cathode but a little way out creating a 'glowing tube' effect around the shaped cathode. The electrons must first be accelerated to the necessary energy level.

The colour of the emission is orange with a gas filling of pure neon, adding some argon to the mix gives the orange a pale violet fringing. Blue effects can be attributed to the addition of mercury vapour.

The series resistor

Current through the Nixie must be limited by using a resistor in series with the power supply. Without the resistor, arcing will occur causing the tube to overheat and be damaged. The series resistor value can be calculated (in the same way you would for an LED) using the formula:

$$R = V / I = (V_s - V_{maint.}) / I_{nom.}$$

A typical value lies in the range from 10 to 80 kΩ. The anode to cathode strike voltage and

Tube-In-A-Tube

Ronald Dekker has a good selection of Nixie-based projects on his website. One of these is a Nixie clock powered directly from the mains mounted in an inverted test tube. The circuit doesn't require any high voltage transistors or special ICs (www.dos4ever.com/TiT/TiT.html).

An engine block clock

This is unusual, a DCF clock mounted in a BMW engine block! Steffen Möritz developed this idea in his spare time and has also designed a rev counter with Nixie tube display for his 1961 Mercedes-Benz.
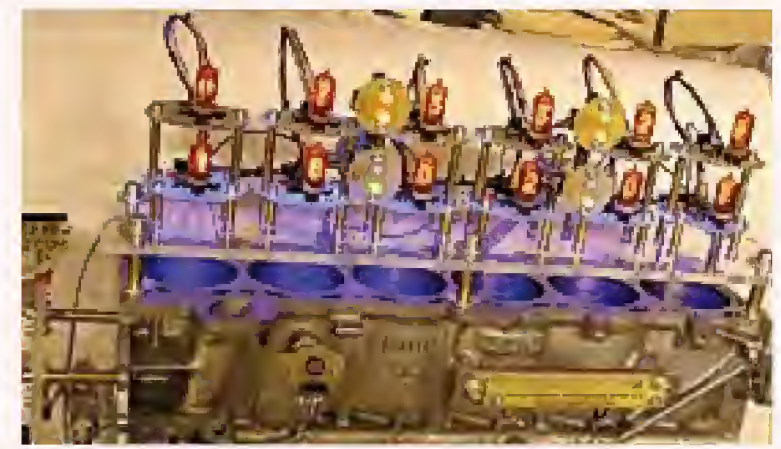
Nixie clock meets Ethernet

Nixie tubes and clock designs belong together. Tobias Krista has designed his with an Ethernet interface. It can request the current time from an NTP server (Network Time Protocol) without the need for a PC. In addition all the functions (Alarm clock, Countdown etc.) can be accessed remotely using a web interface.

A silicon-free clock (display)

This Nixie clock by Michael Pape doesn't use any silicon devices at all. The circuit uses over 100 XC18 type valves. Several ring-counter circuits are used to derive a 1 Hz clock from the AC power frequency.
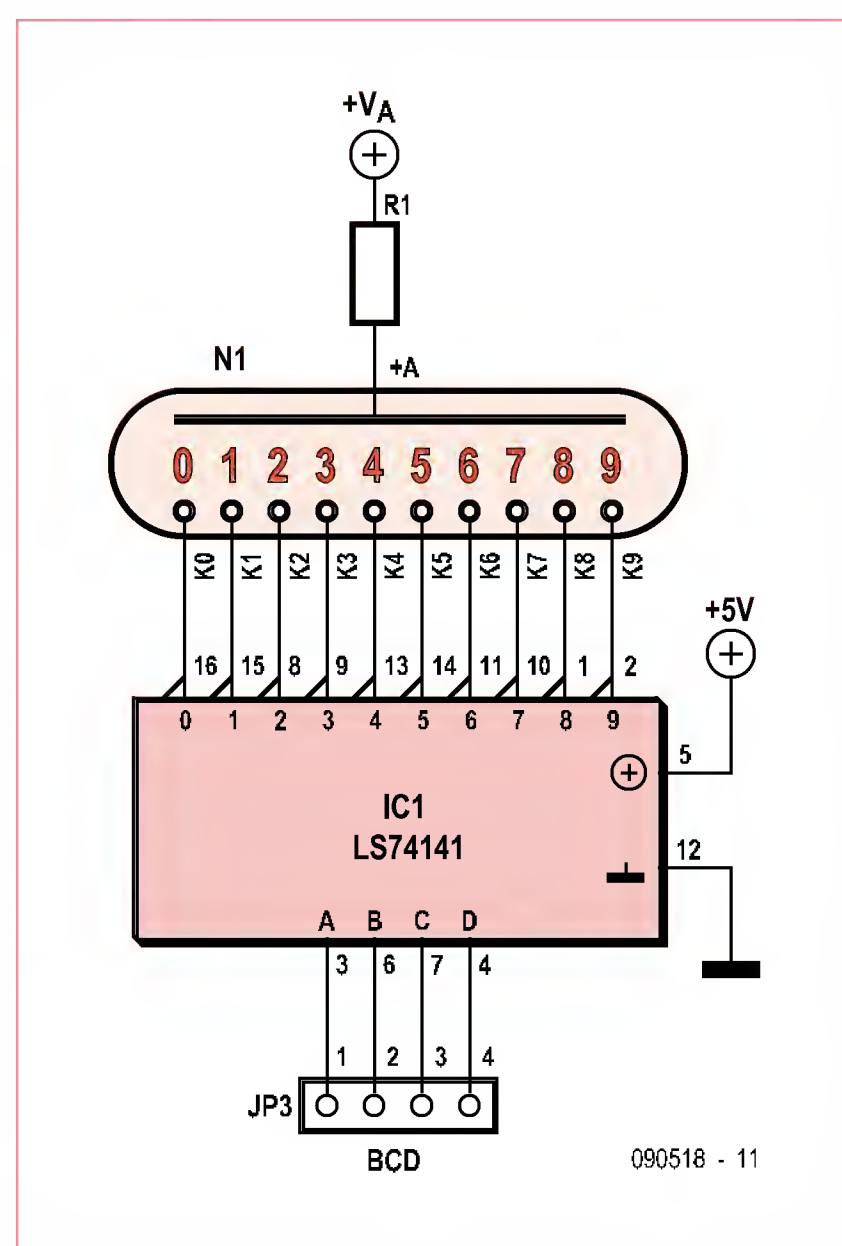
Figure 4. The LS74141 driver IC contains a BCD to decimal decoder and ten transistor drivers.

nominal maintaining current for a particular type of Nixie tube will be given in the corresponding data sheet. Where there is no data sheet available the series resistor value can be ascertained empirically. Using a supply voltage of 180 V identify the common anode pin and cathode pin connected to the displayed character with the minimum surface area (usually the number '1'). Take precautions to ensure that all connections are properly insulated. Connect a high value resistor (a 100 kΩ pot is a good choice here) in series with the supply and tube. Slowly reduce the resistance of the pot until the character starts to glow. When the whole area of the character is glowing uniformly and current through the tube is within spec (1 to 5 mA, or more for larger Nixies) turn off, remove the pot and measure its value. Now select the next highest fixed resistor from the E12 series to use as the series resistor. Larger characters require a smaller resistor (greater area of emission). Take care with the resistor power rating; 2 mA through 100 kΩ will fry a 0.25 W resistor.

Driving the Nixie

The Nixie tube requires a relatively high supply voltage but not too much current. There have been many different systems to drive the device, some of the more common methods are shown here in chronological order.

Unfortunately it is not possible to use LED driver chips directly because they are not be able to handle the high voltage required for the tubes (although [8] is an exception).

Beam switching tubes

Originally a Nixie would have been driven by a 'beam switching tube'; a vacuum tube decade counter which counts using the electromagnetic effect. The fastest versions could be clocked at speeds up to 10 MHz [9].

The tube can interface to a Nixie tube with very little additional circuitry directly. This method was originally used by Burroughs; these tubes are however more scarce nowadays than Nixies. In addition they are quite bulky [10].

Precisely two clocks!

These two clocks are the work of Loek Riemens. The first clock (at the bottom) receives time information via DCF while the second uses GPS satellite data to provide time information. The display of the GPS version changes colour from orange to red when it is receiving a signal from less than two satellites.

A remarkable watch

A wristwatch with Nixie tube displays? Reader Hugo Marien flagged up this project designed and built by Jeff Thomas. This watch measuring 7 x 4 x 3 cm uses miniature Nixie tubes and is sure to turn heads. Go to Jeff's website (www.amug.org/~jthomas/watch.html) for more information.

Nixie voltmeter

Nixies are seldom used in voltmeter designs but this project designed by the author of this article can be found on his website. It uses Russian tubes type IN-14 and IN-19B. (www.jb-electronics.de/html/elektronik/nixies/n_volt-meter.htm).
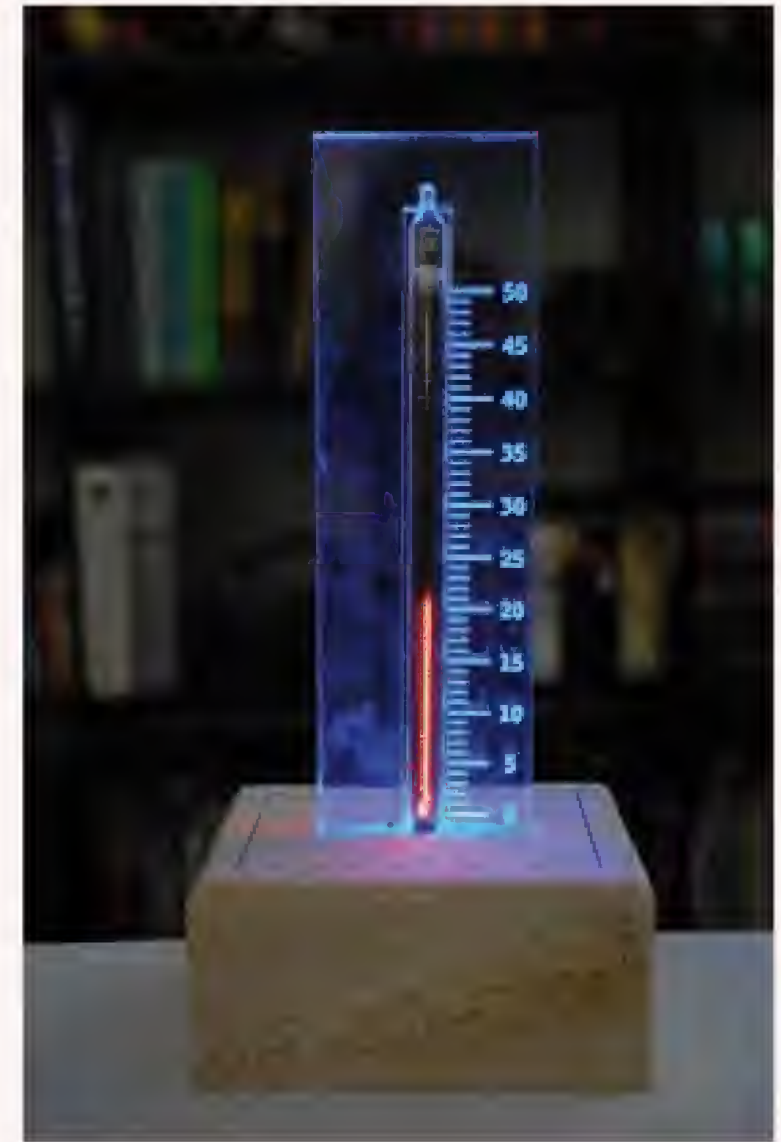
Illuminated room thermometer

This rather elegant room thermometer is the work of Simon Law and Alex Tsekenis. The temperature is displayed using an IN-13-Nixie tube. The current version has both Celsius and Fahrenheit scales and the backlight colour changes according to temperature — awesome!

Discrete Driver ICs

With the advent of digital technology came the driver IC type LS74141 (or the Russian equivalent K155ID1). It is a BCD to decimal decoder together with ten driver transistors in a 16-pin DIL outline [11]. The circuit is given in **Figure 4**. These chips are widely available (eBay or [12]) but their price is steadily rising, currently they cost 1 to 2 €. The advantage of this chip is the space saved on a PCB layout. One disadvantage is that only one output can be active at any time (the BCD to decimal decoder activates 1 of 10 outputs). In addition these chips cannot handle too much current; 7 mA (i.e. 80 mW) maximum.

Transistor drivers

Driving the tube using discrete transistors is also a possibility. The MPSA42 (NPN) and the MPSA92 (PNP) are the most common 'Nixie transistors'. Both of these are available in SMD and THT packages at reasonable cost. The disadvantage of this approach is that a transistor is required to drive the cathode of each displayed element so it consumes a relatively large area of the PCB lay-out (see **Figure 5**). Compared to the driver IC method described above discrete transistor drivers are more flexible and can be used to drive any bit pattern that may be required by the Nixie tube and not just one single element in the tube. Transistors can also drive more current (up to 100 mA), enough for all types of Nixie.

The Supertex HV series

The company Supertex [13] produce chips for high voltage applications for driving displays such as vacuum fluorescent (VFD) and plasma displays, some of these chips are also suitable for use with Nixies.

These devices are more sophisticated than the methods described above; not only can they drive high voltage outputs but they also contain shift registers to store the displayed bit pattern. They therefore have few input signals (Data input, Clock, Strobe and Blanking) but 20, 32 or up to 64 outputs. **Table 1** gives an overview of some of the more interesting ICs.

Some of the chips in the HV family of chips do not allow multiple outputs to be switched simultaneously so make sure you study the relevant data sheets. These chips are not stocked by the usual chip suppliers in the UK. It will be necessary to do some investigative work to track down a source [14].

Supplying the juice

We have already mentioned that these devices require a high voltage supply but
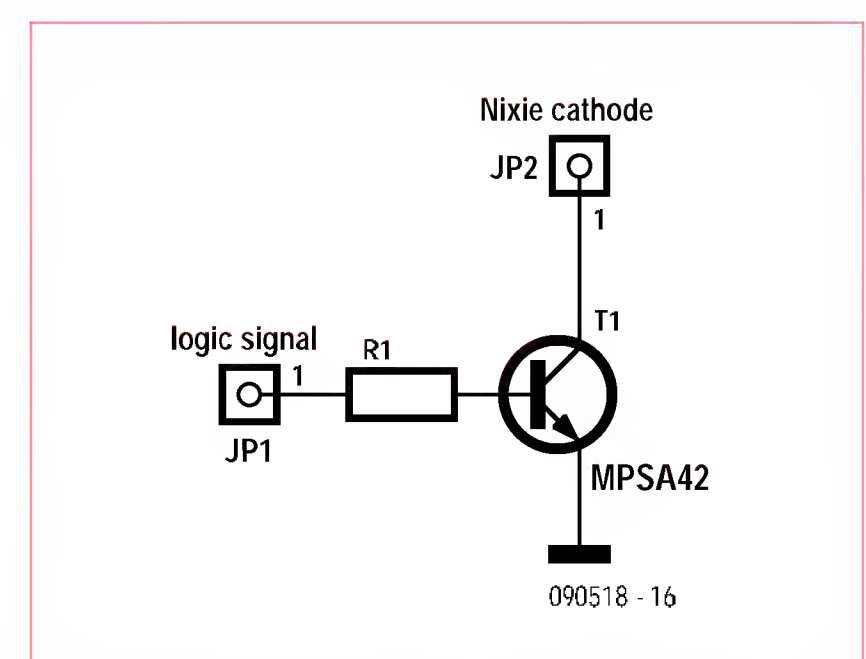
Figure 5. A circuit using discrete driver transistors.
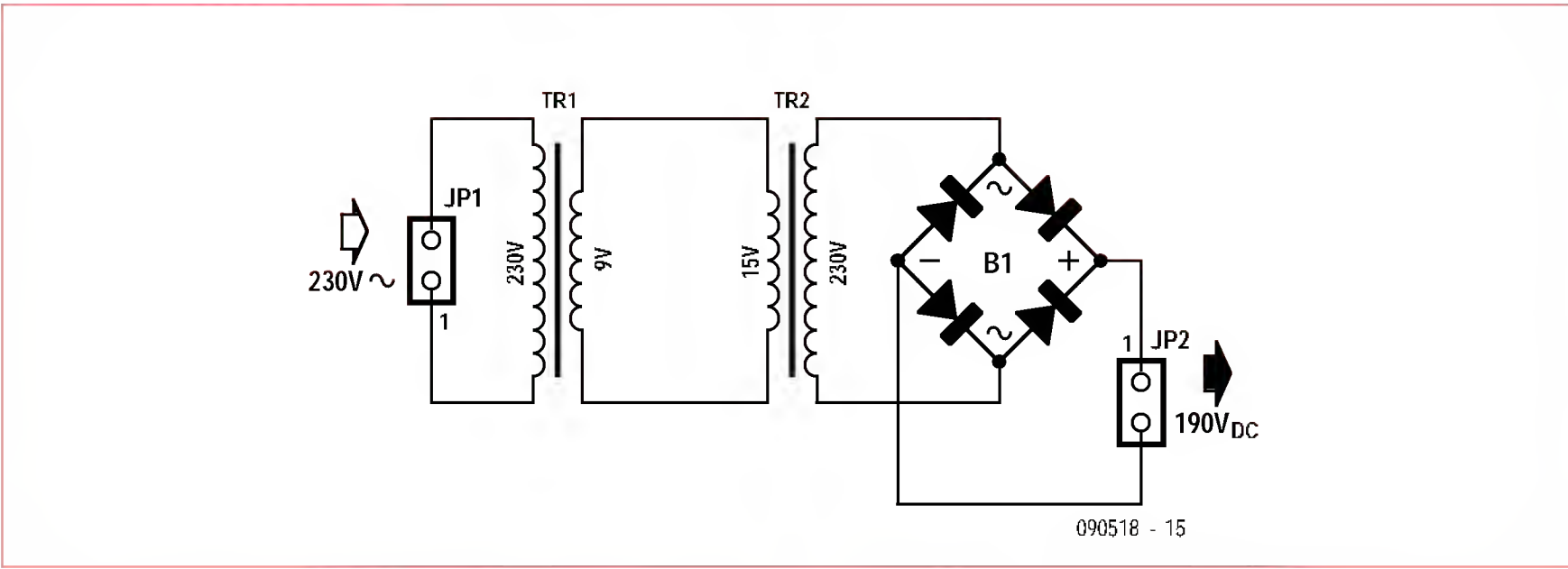
Figure 6. The high voltage is produced by reverse-connected TR2.
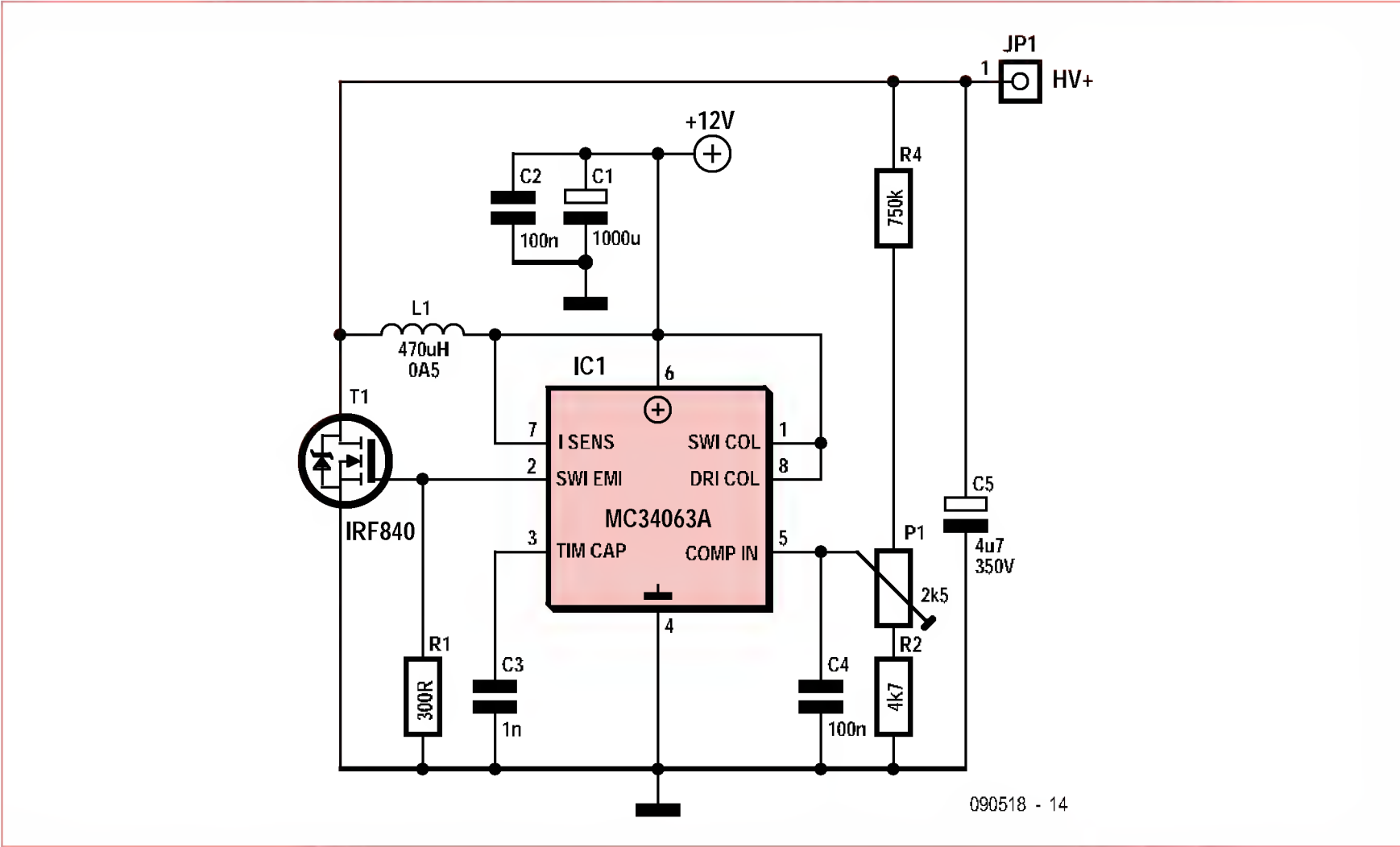
Figure 7. Boost converter using a (Standard) switch regulator type MC34063A [16].

isolation for the entire circuit. The first PCB mounted 9 V transformer is wired to the AC as usual but its secondary is wired to the low voltage winding on the second transformer (see **Figure 6**). Choosing a 15 V transformer for T2 will give an output voltage of 138 VAC at its primary winding. After rectification and smoothing a DC voltage of 190 V is produced — good enough for the majority of Nixies. The maximum output current from this design is limited by the transformer ratings.

A Switch Mode Power Supply (SMPS)
Switch mode power supplies using the Boost-Converter (Step-up-Converter) principle can be used to generate the high voltage. The circuit relies on the principle of electromagnetic induction to generate high voltage. A current is passed through a coil and then abruptly interrupted. The voltage induced in the coil is proportional to the change of magnetic flux. A very fast step change will induce a voltage much higher than the supply voltage to the circuit. The induced voltage is then passed through a diode and smoothed with a high voltage capacitor. When this process is repeated it produces a stable high voltage supply.
In **Figure 7** shows the circuit for boost-converter supply using a standard switch regulator IC type MC34063A (thanks to Dieter Wächter for this circuit [15]).
Ready-made, fully assembled and tested switch mode supplies are also available for the less adventurous, designing a low-noise supply is not a trivial exercise. A good overview of some commercial designs can be found at [16].

not yet suggested how this can be achieved. A number of possibilities exist:

Direct from the AC outlet?
The simplest and least expensive option (as the author has sadly too often witnessed) is to rectify the AC outlet voltage. With this approach the complete circuit will be at AC grid potential and therefore hazardous. You have been warned!

Back to back
A power transformer with a 140 V secondary is a fairly rare beast. A better alternative is to use two low voltage transformers wired back to back. This achieves galvanic

Table 1. High voltage driver					
Name	**Output**	**Voltage**	**Current**	**Outline**	**Price [15]**
HV5522	32 (open drain)	220 V max	100 mA *	44 PLCC, 44 PQFP	5.50 €
HV5630	32 (open drain)	300 V max	100 mA*	44 PLCC, 44 PQFP	6.17 €
HV9708	32 (push/pull)	80 V	20 mA, 5 mA	44 PLCC	4.16 €
HV5812	20 (push/pull)	80 V	3,5 mA	DIP28, 28 PLCC	2.27 €
* check max power dissipation!					

The Author

Jens Boos studied physics at RWTH Aachen, Germany. He includes electronics and collecting Nixie tube as hobbies. He already has over 300 tubes in his collection gathered from all over the world including some rare and valuable examples. For more information visit the authors website

www.jb-electronics.de.

Web links and Literature

[1]   Elektor (German), January and February 1972,
      'Digital Frequency Counter'

[2]   www.elektor.com/050018

[3]   www.elektor.com/090784

[4]   US-Patent US2142106,
      issued on 3rd January 1939 to Hans Paul Boswau

[5]   Serial number '72020699', applied for in 1955, view under
      http://tess2.uspto.gov/

[6]   www.amug.org/~jthomas/clockwork.html

[7]   www.jb-electronics.de/html/elektronik/nixies/index.htm

[8]   http://dos4ever.com/QandA/QandA.html

[9]   www.decadecounter.com/vta/tubepage.php?item=18

[10] www.radiomuseum.org/forum/
      nixie_and_trochotron_haydu_vs_burroughs.html

[11] www.tube-tester.com/sites/nixie/74141-NDT/74141-NDT.htm

[12] www.askjanfirst.de

[13] www.supertex.com/

[14] www.mouser.com/

[15] www.nixie.tube-tester.com

[16] www.tayloredge.com/storefront/SmartNixie/PSU/
      comparison.html

[17] www.nocrotec.com

[18] www.kosbo.com

A bright future for Nixies

There is no doubt that Nixies are of interest not just to collectors of electronic curiosities but also to home brewers worldwide. The following page shows some fascinating Nixie based designs. The enduring appeal of them is evident from their rising prices on the second-hand market. The tubes are now changing hands at four or five times the price they were fetching at the beginning of the year 2000. A good selection can usually be found on eBay or from specialist websites ([12] [17] [18]). Whether the existing stocks of Nixies will be sufficient to cater for the growing interest in this technology remains to be seen.

(090518)

Audio DSP Course (1)

Part 1:

Audio signal processing with a DSP

Since the advent of digital signal processing around 30 years ago, audio signal processing has increasingly shifted from analogue technology to digital technology. In this course, in addition to introducing you to the properties of digital signal processors for audio signals and the associated programming aspects, we present several applications based on an inexpensive but nevertheless high-performance signal processing module.

By Alexander Potchinkov (Germany)

Just as an operational amplifier is a general-purpose component for the analogue processing of audio signals, a digital signal processor (DSP) is a general-purpose component for the digital processing of audio signals. However, this comparison tells only part of the story, since a digital signal processor — besides having more pins than an operational amplifier — is considerably more complex and can do a lot more. This series of articles begins with a brief discussion of digital processing of analogue audio signals. Next we present and describe the key features of the Freescale DSP56374 digital signal processor, which comes in a 'soldering-friendly' 52-pin package and is specifically designed for the task of digitally processing audio signals. With a processor clock frequency of 150 MHz and sophisticated signal logic, it provides enough computing power to implement very complex audio signal processing tasks.
We also present three practical projects based on a DSP board developed for this series of articles:

- A digital audio signal generator that can output low-distortion sine-wave signals, white and pink noise, as well as octave band or one-third octave band noise
- A peak-value signal level meter with LED bargraph display
- A dynamic signal processor with compression, limiting and noise gate functions

Digital processing of analogue audio signals

A microphone generates an analogue signal, and loudspeakers are driven by analogue signals. For many decades, audio technology was analogue all the way from the microphone to the speaker. High-fidelity technology in the period from the late 1960s to the 1980s was also analogue, and as many of us know, it achieved highly satisfactory results. CDs as replacements for phonograph records first appeared on the scene in 1982, and they left phonograph record far behind on the cost and quality fronts. At first, relatively little digital

audio signal processing was needed for CD technology. In simplified terms, all that was necessary was to use an analogue to digital converter (ADC) to digitise the analogue signal from the microphone, copy the resulting digital data to a CD, read it out in a CD player, and use a digital to analogue converter (DAC) to change it back into an analogue signal. From an external perspective, this is a signal path with analogue input and output interfaces.

As you know, CDs work with 16-bit word resolution. It is not especially easy to produce converters (ADCs and DACs) with suitable quality. This relates not only to the converters themselves, but also to the analogue filters necessary for their operation: the anti-aliasing filter ahead of the ADC and the reconstruction filter after the DAC. The engineers at Philips who developed the CD technology recognised this early on, and they came up with a clever idea for sophisticated digital signal processing. In simplified terms, Philips managed to achieve nearly 16-bit quality with a 14-bit DAC by using oversampling and noise shaping in an upstream digital filter, which amounted to genuine digital signal processing.

In this connection we should mention a fundamental problem in the digital processing of audio signals. Analogue audio signal processing has the highly desirable property that nonlinear distortion is low at low signal levels and high at high signal levels. This matches our sense of hearing, since we hear better at low acoustic levels than at high acoustic levels. Unfortunately, digital audio signal processing has exactly the opposite property. At low signal levels the amount of nonlinear distortion is high relative to the desired signal, while at high signal levels it is relatively low. This has inspired engineers and mathematicians to approximate the analogue situation by devising methods that reduce the perceived amount of nonlinear distortion at low signal levels or allow the distortion to be filtered out of the audible frequency range relatively
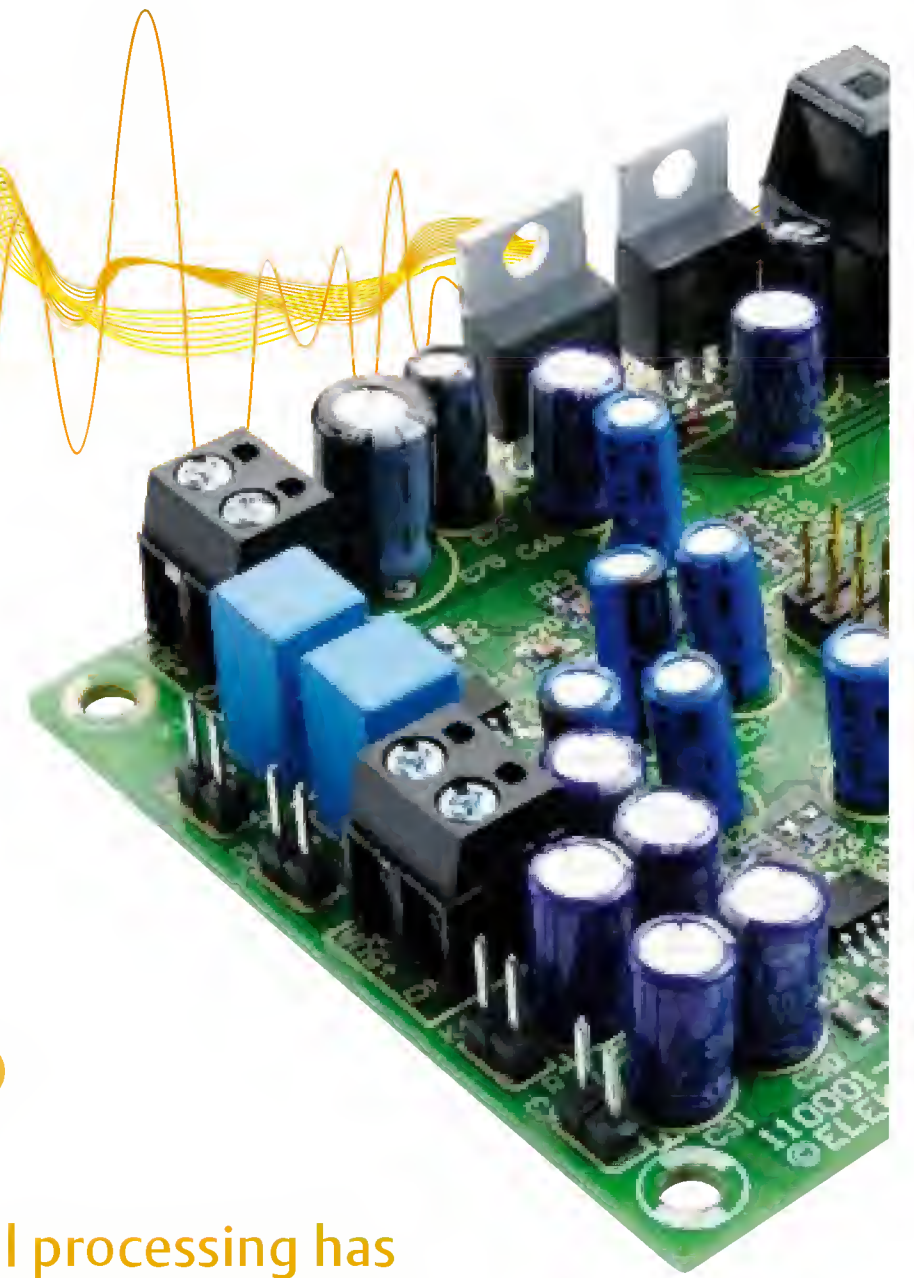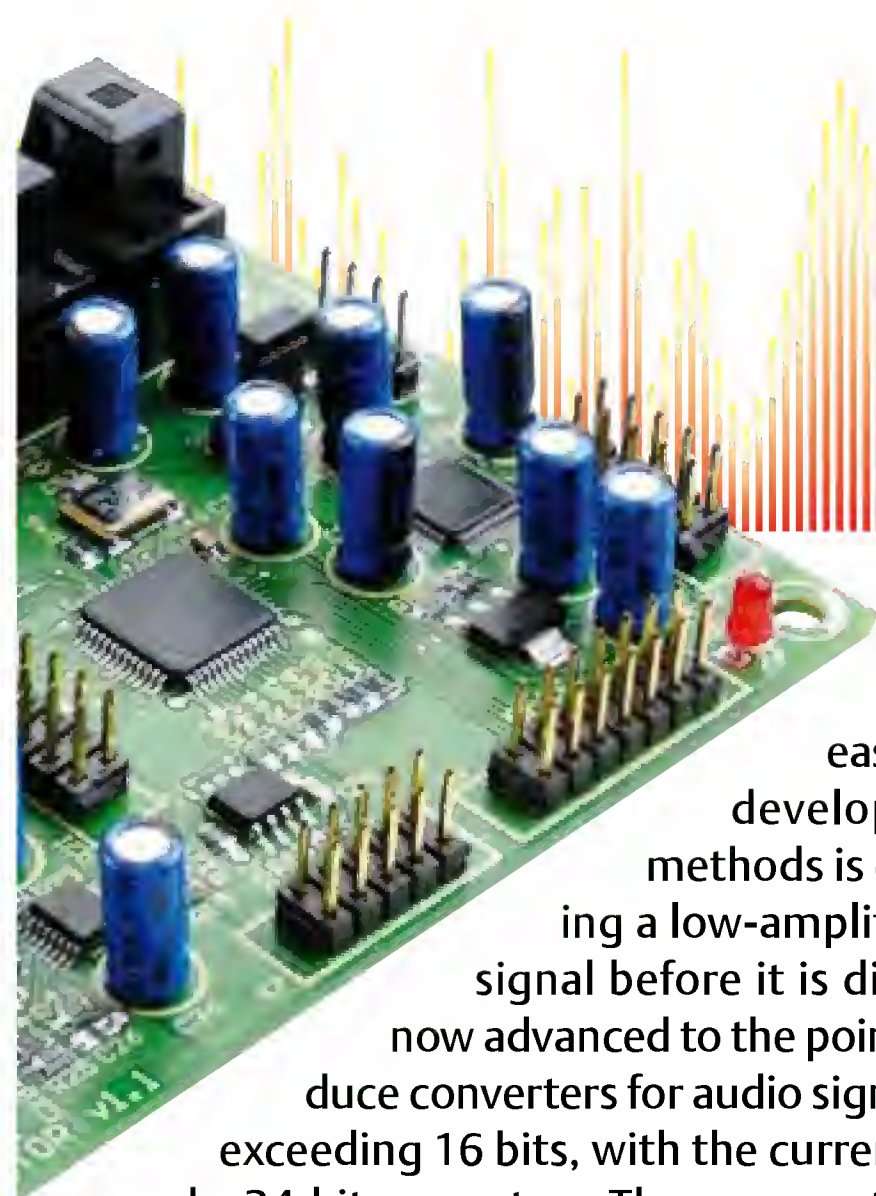
easily. Along with the techniques developed by Philips, one of these methods is dithering, which involves adding a low-amplitude noise signal to the audio signal before it is digitised. These methods have now advanced to the point that they can be used to produce converters for audio signal processing with resolutions exceeding 16 bits, with the current state of the art represented by 24-bit converters. These converters employ almost exclusively digital techniques for audio signal processing and require only a little bit of supplementary analogue signal processing.

Digital signal processors

A digital signal processor is a special type of processor designed to provide many functions that fulfil the typical requirements of digital signal processing. This involves the processing of repetitive algorithms, usually with fairly simple structures but a large number of computation steps, under real-time conditions. The clock rate for executing the algorithms is determined by the sampling rate of the digital data, which in the case of a CD is set by the sampling frequency of 44.1 kHz. This means that the DSP algorithm must be fully processed in approximately 22.7 μs, from which the time necessary for servicing the audio interfaces must be subtracted.

Processor architecture

Figure 1 shows the block diagram of the DSP56374. It has a typical Harvard architecture, with multiple data and address busses and several memory areas that can be accessed in parallel. DSPs are register-based processors. The operands and the results of the computations, most of which are arithmetical, are stored in registers or read from registers. This enables single-step operations without microcode, but it also requires load and store transactions between the memory areas and the registers. Fortunately, the hardware architecture and the address generation units (AGUs) allow this to be done in parallel with the arithmetic computations. AGUs are computation units that convert memory addresses into typical DSP addressing modes independently of the actual data processing.
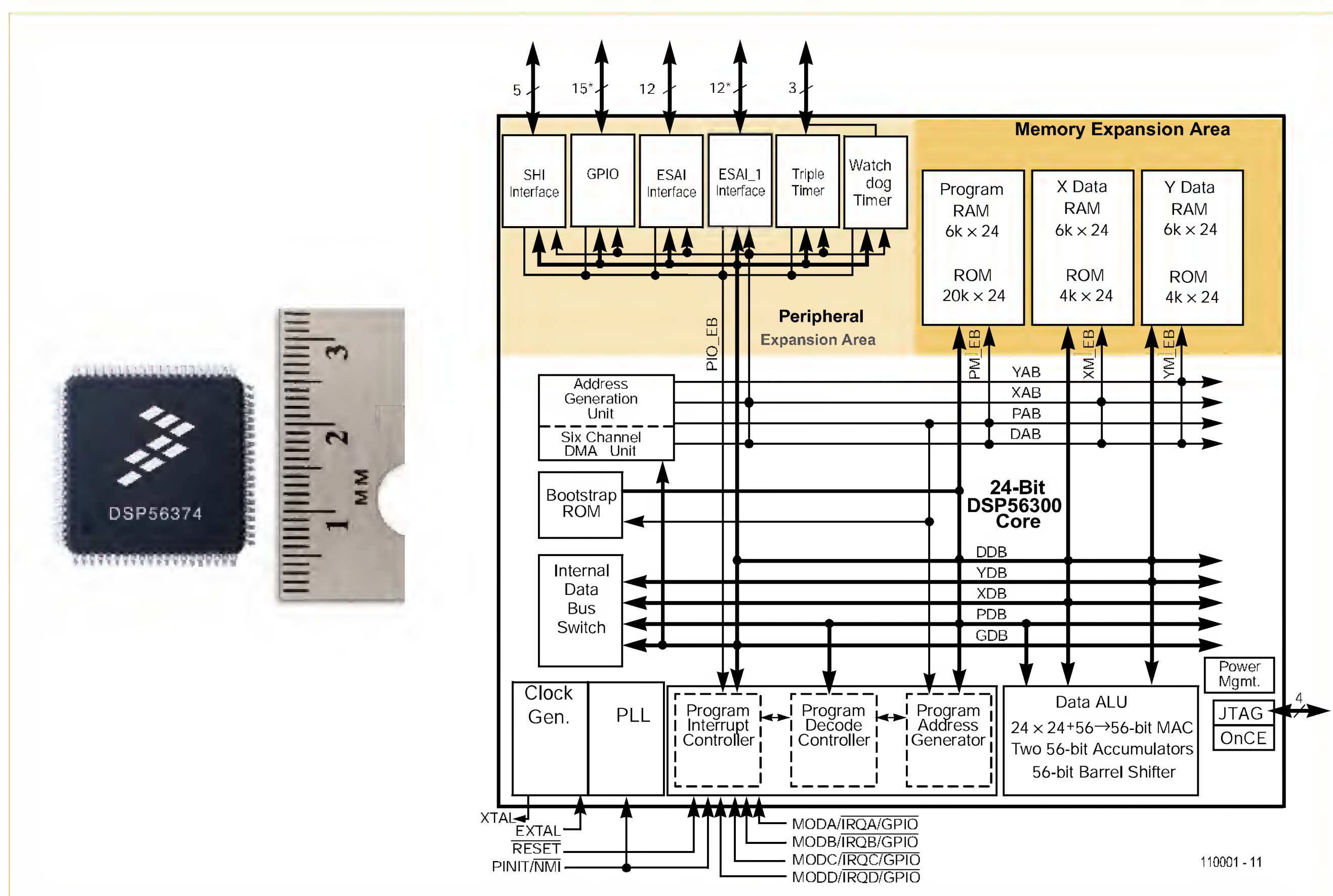
Figure 1. Block diagram of the DSP56374 digital signal processor.

The Serial Host Interface (SHI) block is a synchronous serial interface that can be operated in SPI or I²C mode. In both of these modes, the DSP is able to act as a bus master or as a bus slave. This interface can be used for connection to a microcontroller, among other options. The General Purpose Input/Output (GPIO) block provides I/O ports for general tasks, similar to the I/O ports found in just about ever microcontroller. Each of the two Enhanced Serial Audio Interface
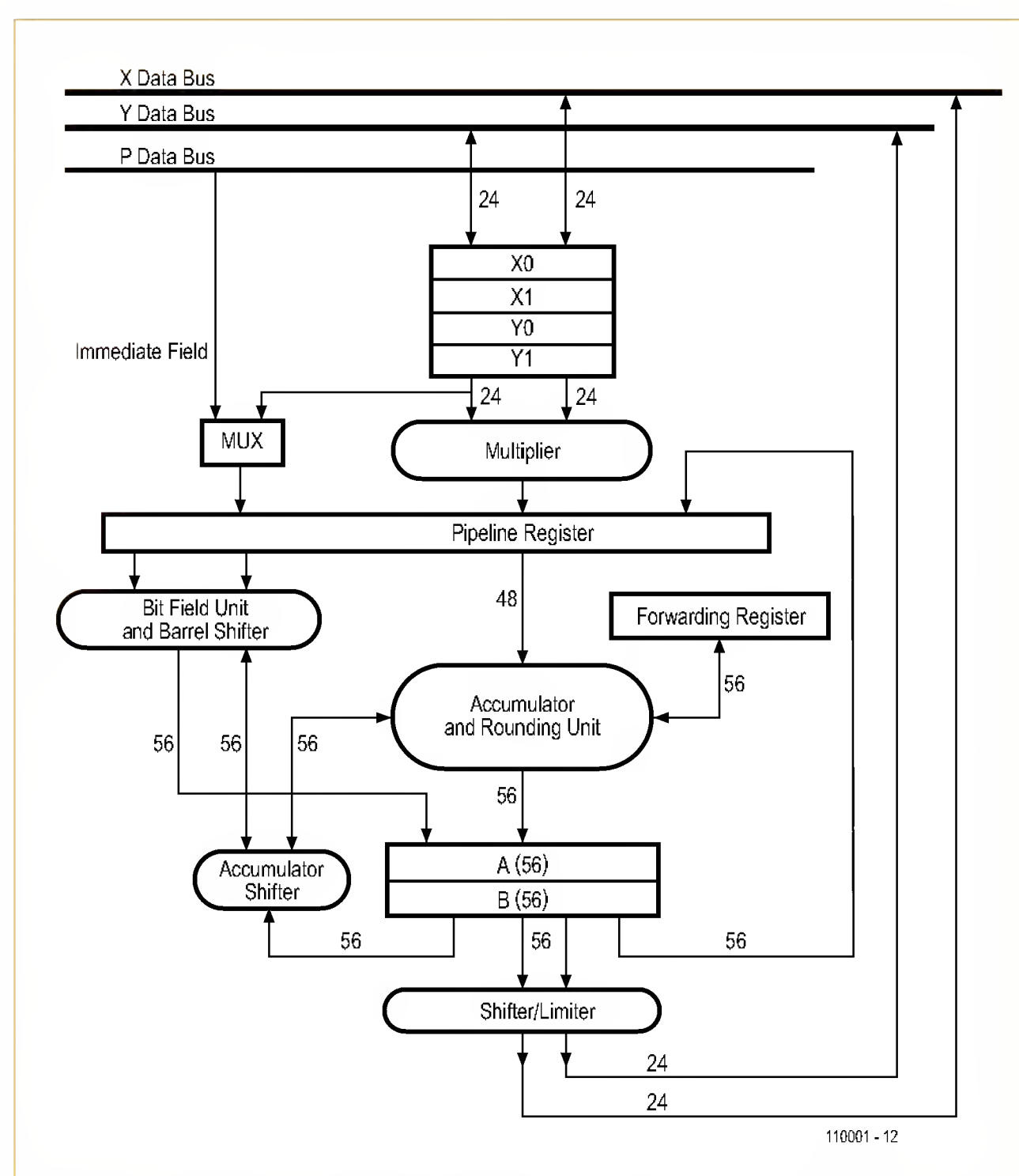
Figure 2. Data path of the DSP56K 'Symphony' family of audio data processors.

blocks (ESAI and ESAI_1) provides six audio data ports together with the ports necessary for synchronisation. The ESAI_1 block is omitted in the 52-pin DSP56374 version. The outputs of the three timers in the DSP and a watchdog timer, which can be used to check whether the DSP is still running or has crashed, are fed out over three ports. The Memory Expansion Area group contains three banks of RAM, comprising X and Y RAM for data and program RAM (P RAM) for executable code. The ROM portions of these memory areas are not significant for our purposes. The sizes of the individual memory partitions (X, Y and P) are configurable.
The AGU block contains two high-performance, versatile processing engines that can simultaneously compute two addresses in a single processor clock cycle. The bootstrap ROM contains the code for booting the DSP. For example, the DSP can be booted serially over the SHI port. The Clock Generator and PLL blocks jointly gen-

FIR filters with the DSP56374

The number format used to represent samples in the DSP is called twos-complement fractionals, or simply fractionals. The filter coefficients, which are also fractionals, are stored in DSP memory as 24-bit words. They define the characteristics of the filter.

There is a certain amount of filter design software available, which can be used to design both finite impulse response (FIR) and infinite impulse response (IIR) filters. Both types of filter employ signal feedback loops and are suitable for the direct implementation of well-known analogue filters.

Figure 3 shows the signal flow diagram of five-stage filter. This is a graphic representation of the following differential equation:

$$y(n) = b_0x(n) + b_1x(n-1) + b_2x(n-2) + b_3x(n-3) + b_4x(n-4),$$

which specifies the computation rule of the filter.

The square box labelled z^{-1} represents the storage of a 24-bit sample. The triangle symbol represents the multiplication of a sample by a 24-bit factor, which in this case is a filter coefficient. The circle with a plus sign (+) represents the addition of two 48-bit numbers.

At sample time n the time window contains five samples of the input signal: $x(n)$, $x(n-1)$, $x(n-2)$, $x(n-3)$ and $x(n-4)$, which for example may originate from an ADC with 24-bit word width. The value $x(n)$ is the current sample, while the values $x(n-k)$ are the previous samples obtained k sample intervals in the past. The calculation of the current output value $y(n)$ requires five MAC operations as well as the initialisation of the accumulator to zero. Accordingly, the following six steps must be executed, and a counting loop can be for this purpose:

```
a := 0;
a := a + b0 * x(n)
a := a + b1 * x(n-1)
a := a + b2 * x(n-2)
a := a + b3 * x(n-3)
a := a + b4 * x(n-4)
```

erate the processor clock signal. The DSP can be used with a low-frequency external clock generator, whose signal is multiplied to achieve the processor clock frequency with a maximum value of 150 MHz. The Program Interrupt Controller block handles the numerous internal interrupt options. Many interrupt sources are available from the interfaces. Next to it is the block that controls the command sequences. To its right there is the Data ALU block, which is the data path. This is where the digital signal processing is executed. Finally there is the JTAG/OnCE block, which can be used to download and debug DSP programs with the aid of a PC-based program (debugger).

Data path

The data path is shown in **Figure 2**. The core of the data path is the MAC unit. 'MAC' stands for 'multiply and accumulate', which is a
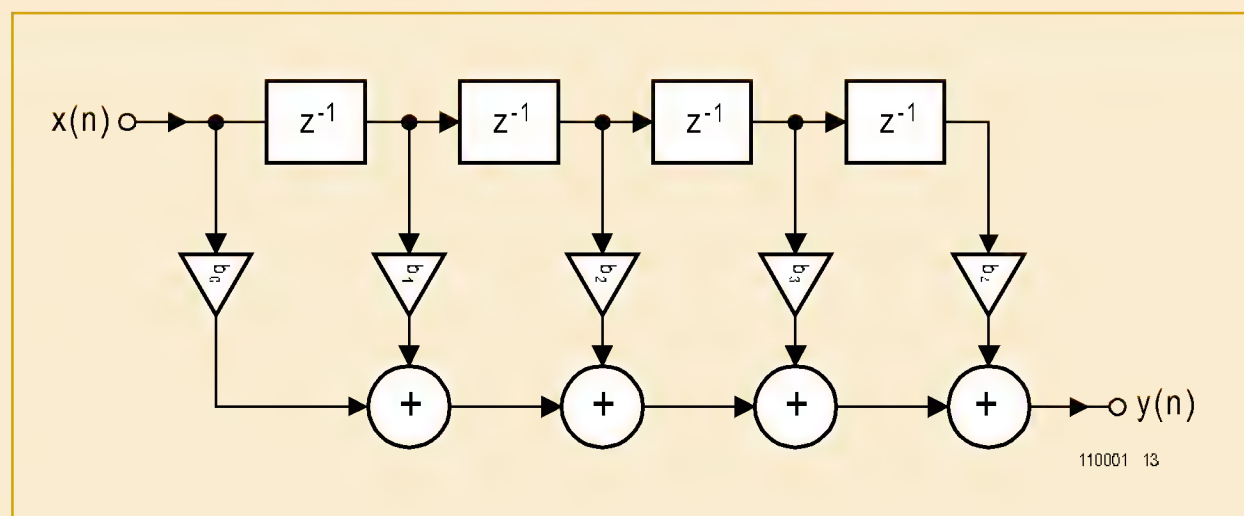
Figure 3. Signal flow diagram of an FIR filter.

Before we describe the DSP code, we need to say a few words about the two memory areas needed for this purpose. In X RAM we use linear addressing to store the coefficients in a memory area with length $n = 5$ and an arbitrary base address. The data for the current time window is always stored in Y RAM. Ring buffer addressing in achieved by using modulo n addressing. In the following code, *cbase* is the base address of the coefficient memory area and *dpntr* is a memory location in Y RAM that holds the data area pointer between calls to the filter routine. At the start of the process this memory location holds the base address of the data storage area.

```
          move    #N-1,M4                        ; modulo N indexing
          ...
Audioloop
          ...
          movep  x:RX0,y1                        ; read audio sample
          move   #cbase,R0                       ; set coefficient pointer
          move   y:dpntr,x0
          move   x0,R4  ; load data pointer
(1)       clr    a    x:(R0)+,x0    y1:y:(R4)+    ; read 1st. coeff, save 1st data
(2)       rep    #N-1
(3)              mac  y1,x0,a    x:(R0)+,x0   y1:y:(R4)+

(4)       macr   y1,x0,a       (R4)-            ; requantisation
          move   a,x:TX0                         ; write audio sample
          move   R4, x0
          move   x0, y:dpntr                     ; save data pointer
          jmp    Audioloop
```

In instruction (1), the accumulator is initialised to 0 concurrently with two data transfers and pointer incrementing (using modulo arithmetic in the case of the *R4* pointer). Filter coefficient b_0 is written to register *x0*, and the current audio data (sample) is written to the ring buffer. For the next *mac* instruction, this filter coefficient can be used along with the current audio data, which is located in register *y1*. Instruction (2) initiates a hardware counting loop whose body is called and executed *N-1* times. Loop instruction (3) computes and accumulates a partial product, while at the same time writing the coefficient and data for the next partial product to registers *x0* and *y1*. The result of the accumulated partial products is held in the accumulator, which has a word width of 56 bits. The *macr* command in instruction (4) performs the final partial product computation and accumulation operation and requantises the 56-bit result to 24 bits as required for the DAC. The pointer to the data area is decremented by 1 at the same time.

computation with the form $a := a + x*y$. This is one of the key computations in digital signal processing. Among other things, it is used to implement digital filters. An accumulator holds the product $x*y$, which requires the services of a multiplier, an adder and an accumulator register with suitable connections and word widths. As you can see in the figure, the two operands used to generate the product can be loaded over two data busses (*X* and *Y*) into any of four 24-bit registers (*x0*, *x1*, *y0* and *y1*) forming a small register file.

The DSP has two accumulator registers (*A* and *B*), which are connected to the busses. This avoids the need to store the intermediate results of an accumulation operation in memory. When a series of such MAC operations must be performed in a digital filter algorithm, the DSP allows the samples and filter coefficients for the next computation step of the following MAC operation to be fetched while

the current MAC operation is being executed. As you can see from the figure, the DSP performs computations with 24-bit data, but the accumulators are 56 bits wide. This provides enough room to hold 48-bit multiplication results without any bit loss. as well an additional 8 bits for left-hand number range extension, which is especially useful in accumulation processes.

The operand registers are connected over two 24-bit paths to the hardware fractional multiplier, which performs each multiplication operation in one processor clock cycle. The 48-bit multiplication results can be processed further in the Accumulator and Rounding Unit of the ALU or written to DSP memory over the data busses. The Accumulator and Rounding Unit is used primarily for accumulation, and the combination of this unit and the multiplier forms the MAC unit. The computation engine of this unit can also be used

About the author

Alexander Potchinkov holds the Chair of Digital Signal Processing at the Technical University of Kaiserslautern (Germany) and runs an engineering consultancy specialising in audio signal processing. Aside from DSPs and DSP algorithms, his interests include valve amplifiers and SPICE simulations.

for the addition or subtraction of samples or for logical operations (AND, OR and EXOR). The Bit Field Unit and Barrel Shifter perform bit-string operations and multi-bit shifts, which are also executed in a single processor clock cycle. The Shifter/Limiter below the accumulator registers in the figure is used for tasks such as requantisation, which means conversion of multiplication results to the 24-bit word width of the DSP. Finally, we should mention that the DSP can also perform longword computations, for which purpose two pairs of 24-bit registers (*x0* & *x1*, *y0* & *y1*) are combined into 48-bit registers (*x* and *y*). In this case the instruction

```
add x,b
```

results in a 48-bit addition, while the instruction

```
add a,b
```

results in a 56-bit addition.

There is also something you need to know about programming the data path. The `mac` and `mpy` instructions (`mpy` stands for 'multiply') are what are called 'three-address instructions'. An instruction such as

```
mac x0,y0,a  or  py x1,y1,b
```

specifies three addresses: the addresses of the two operand registers and the address of the result register. The addition and subtraction commands are two-address instructions. In case of a instruction such as

```
add x0,a  or  sub y0,b
```

one of the two operands must be loaded from one of the two accumulator registers before the command is executed, since it is not possible to connect both inputs of the ALU in the Accumulator and Rounding Unit to the operand registers. The operand in the accumulator register is overwritten by the result of the computation. Due to this loading operation, processing of a two-address instruction requires the execution of two commands.

Special computation techniques, such as saturation arithmetic (saturation at the maximum or minimum end of the number range), allow the DSP to mimic the overdrive characteristic of analogue devices. With other types of processors, this characteristic must be emulated in software.

Control logic and address generator

DSPs have special control logic to boost their throughput. In this connection. one of the major features is 'zero overhead loop' hardware, which allows counting loops to be executed with virtually zero instruction overhead. In particular, this means that configuring the hardware loop takes only a few processor clock cycles and the body of the loop contains only signal processing instructions. A hardware loop with only one instruction (such as an arithmetic operation on signal data, possibly with concurrent data transfer) is needed for the FIR filter whose DSP code is described in the **inset**. With this arrangement the loop initialisation overhead is especially low, allowing virtually all of the processing power of the DSP to be used for the FIR filter. With a sampling rate of 48 kHz and a processor clock rate of 150 MHz, FIR filters with lengths up to approximately 3000 can be implemented. In addition to the previously mentioned capability for parallel data transfers, this requires the ability to perform memory addressing in the background without using the data path. The DSP has eight pointer registers (*R0* to *R7*) available for indirect addressing. The type of address arithmetic and the offsets for address incrementing and decrementing are specified in two auxiliary registers for each pointer, which are called the Modifier Register and the Offset Register (otherwise known as the *M* and *N* registers). All 24 registers in the AGU are 24 bits wide, and they can also be used to hold intermediate results or to specify the lengths of counting loops. Values for incrementing or decrementing the pointer can be entered in the *N* register. For example, samples located *n* intervals in the past can be read from a ring buffer to implement a signal delay, which can be used for purposes such as compensating for propagation time differences between the drivers in a bi-wired speaker box and the listener.

TheM register is used to specify the type of address arithmetic that is used. In addition to linear addressing, the DSP can perform modulo addressing or reverse-carry addressing, which are commonly used in DSP applications. Modulo addressing is needed for the implementation of ring buffers, which hold sequential sets of samples used in digital filters and are updated at the sampling clock rate. Modulo-*n* arithmetic is limited to the *n* numbers from 0 to *n*−1, which can be imagined to be arranged on the perimeter of a ring.

Interrupts

Hardware interrupts can be initiated by the DSP processor core, by the DSP interfaces, and by external sources. Software interrupts can be initiated by a running DSP program. In many cases interrupts can be assigned selectable priority levels in order to distinguish between important and less important events.

A major difference between digital signal processors and general-

purpose processors is that the entries in the interrupt vector table are not the start addresses of the interrupt routines (ISRs), but instead two instructions. This allows two classes of interrupts to be defined:

- fast interrupts, consisting of one or two instructions without a return instruction;
- long interrupts, which jump to an ISR (which may be fairly extensive) that is executed in place of the calling program.

Fetching data from one of the DSP interfaces is an especially important DSP transaction. This requires a suitable *move* instruction with the target address in DSP RAM, which can be executed as a fast interrupt. This eliminates the need for saving the processor state on the stack, which is otherwise necessary with an interrupt.

DSP56374 audio interfaces

The audio interfaces comprise six control and synchronisation ports as well as six data ports. Different transmit and receive clock rates may be used. One of the options is the I²S interface protocol, which is the best known audio data format. Up to six audio outputs or four audio inputs can be used, as long as the total number of inputs and outputs does not exceed six. The programmer simply sees six 24-bit registers on the transmit side, into which the audio data to be output is written at the sampling clock rate of the DSP program, along with four 24-bit registers on the receive side, from which data is read by the DSP program.

Coming up next...

This description of the internal architecture of the DSP concludes the first instalment of our audio DSP course. In the next instalment we'll turn our attention to programming the DSP563xx family

(110001-I)

Internet Link

www.freescale.com/webapp/sps/site/
    prod_summary.jsp?code=DSP56374

Microphone Conferencing System
For more effective online meetings
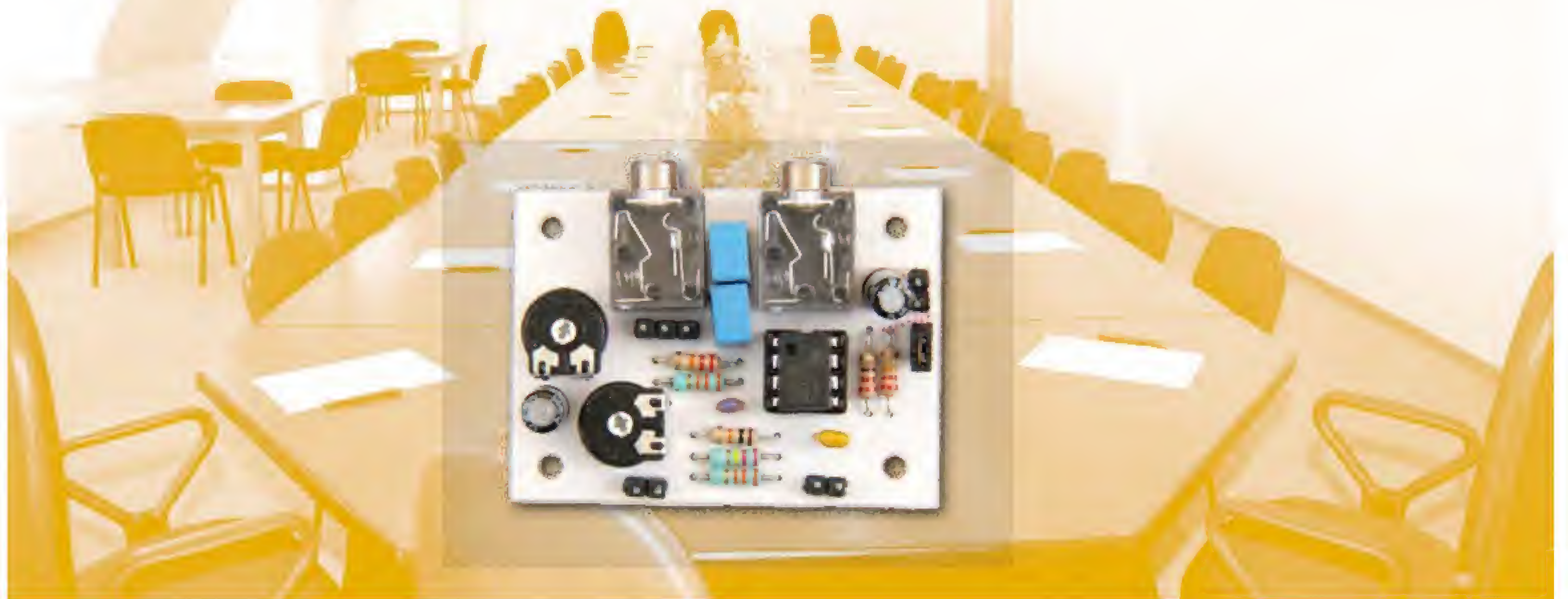
The scope of the online world is forever increasing, while the perceived distances are forever shrinking. (Video) chatting with the children (or the parents) who live far away, international meetings were the manufacturing department in Taiwan can talk via an online connection to the head office in London… It is becoming increasingly commonplace. For meetings that do not involve too many people the microphone built into the webcam or laptop is usually sufficient, but for larger groups or in acoustically less than ideal rooms (with a lot of reverberation, for example) the degree of intelligibility at the other end of the line is often very low. To solve this problem we designed a simple circuit.

By Thijs Beckers (Elektor Netherlands Editorial)

An increasing number of companies make use of online (video) conferencing. This includes Elektor. The intelligibility can be a problem with this. When there is a large group behind the one (laptop) microphone there is the potential that your online colleagues have great difficulty in following the conversation. The quality of the microphone plays an important role, of course,

Characteristics

- Can be used with any PC or laptop with Line or Mic input.
- Simple to operate
- Two units are easily connected
- Pickup pattern of the microphones is easily adapted to suit the situation
- Easy to build using standard components

but in larger rooms, which are not acoustically optimal (lots of reverberation) it is a good idea to use multiple microphones. And Elektor wouldn't be Elektor if we weren't going to have an attempt at solving this ourselves. This article describes our first result: a simple meeting system.

What does it do?

So we are working with microphones. Or to be more accurate: electret capsules, because these are easily obtained and are not nearly as expensive as a 'real' microphone. In addition, they are much more suitable for our intended purpose. And for this reason:

For this application, we thought that it would be good idea to use a so-called 'figure-of-8' recording pattern. We can then position the microphone in the middle of the meeting table where it can be used by four people simultaneously, while the remainder of the incoming sound (reverberation and the like) is rejected as much as possible (see **Figure 1**). We use a trick to do this: by using two back-to-back connected microphone capsules, each having its own omnidirectional characteristic [1], and by amplifying their electrical signals inverted from each other, we obtain the desired recording pattern.

This design has already been described in Elektor March 2003 [2] where we also discussed the distance between the two capsules. This is not allowed to be too large, mainly because the distance between the membranes determines the corner frequency above which the gain will begin to decrease. When this distance is equal to half the wavelength of the frequency then the electrical signals from the capsules are shifted exactly 180° in phase with respect to each other. The difference amplifier will then amplify this signal even more. As the frequency increases and as a consequence
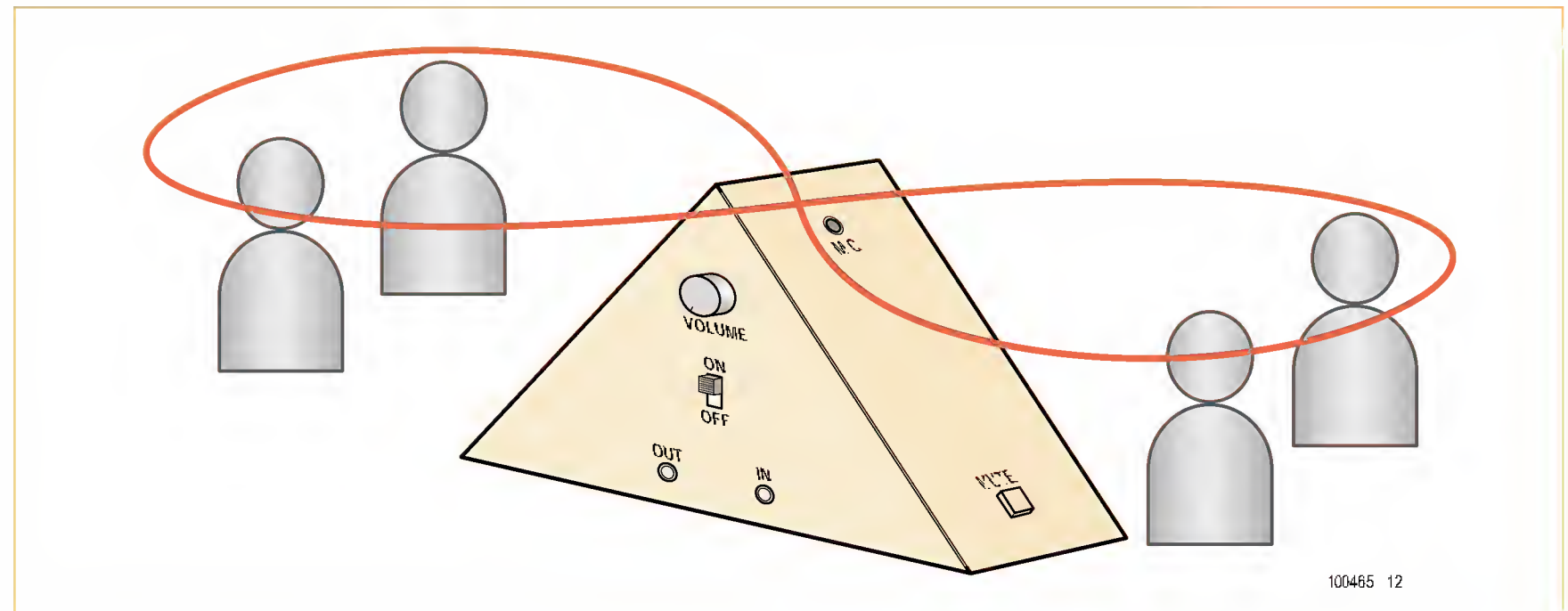
Figure 1. A 'figure-of-8' microphone pattern is created not only to ensure that the registered sound is actually from the people around the table, but also to suppress room noise and reverberation.
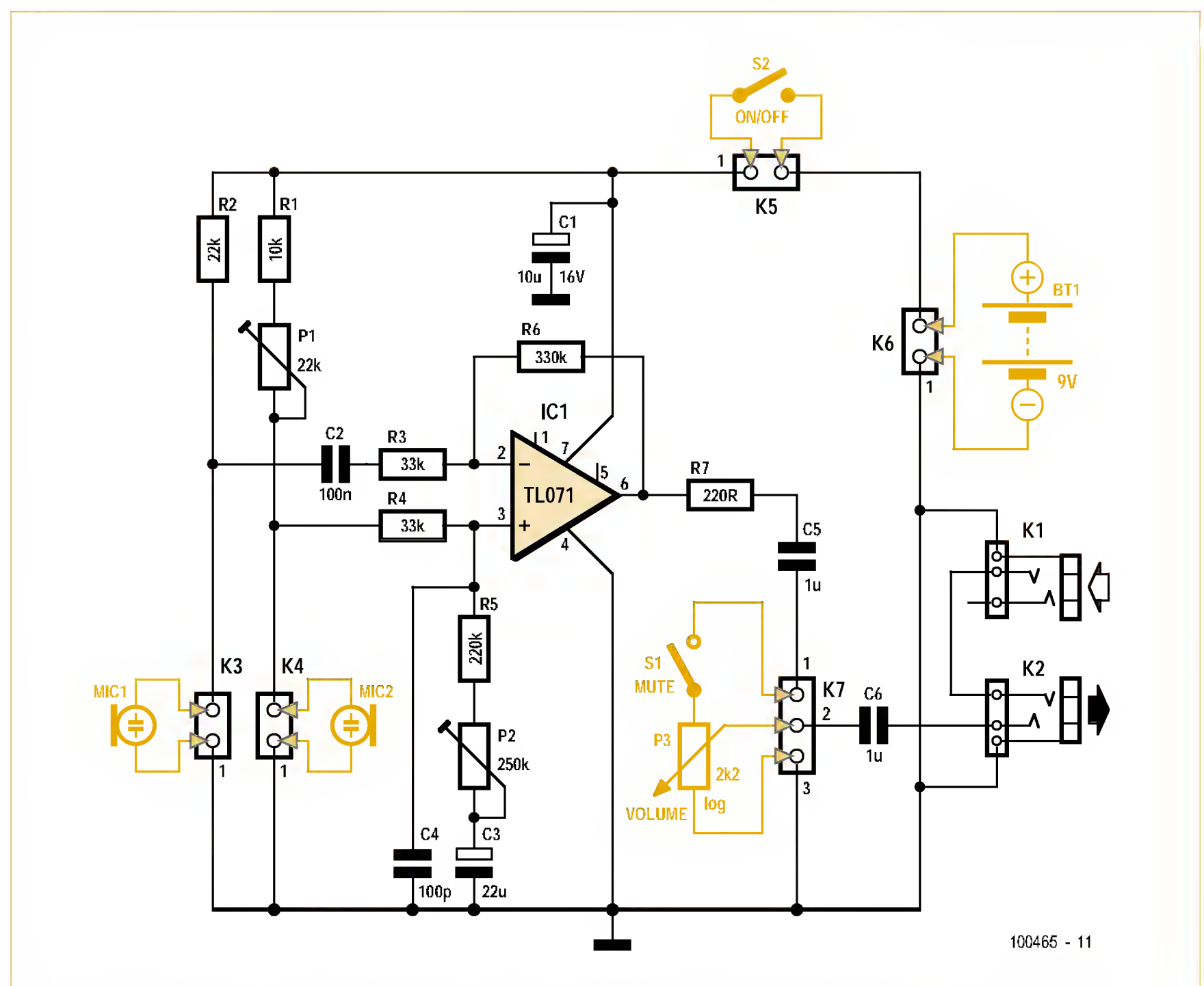
Figure 2. The schematic for our meeting system only contains the essential parts. It doesn't get much simpler than this.

Elektor Products & Services

- PCB # 100465-1
- PCB layout files (pdf and Eagle): # 100465-1.zip
- Hyperlinks in article
- All items accessible via www.elektor.com/100465

is oddball

We haven't tried all available chat and meeting software, but there appear to be differences in the way the various programs use the sound signal. Windows Live Messenger nicely sums the left and right input signals, while Skype only propagates the left signal. In our circuit this would then only transmit one of the microphone signals, while we are offering two.

So that it is nevertheless possible to use two microphone modules with Skype we have made —— in the spirit of the design of this circuit — a rudimentary 'Skype cable'. Figure 3 shows how we did this.

Using two 470 ohm resistors we superimpose both signals and then send them combined through the cable. At the PC or laptop end of the cable we connect the combined signal to the left input only.

As a consequence, each of the circuits do affect one another a little, but this is hardly noticeable in practice. For the sake of simplicity we think this workaround is an excellent solution.
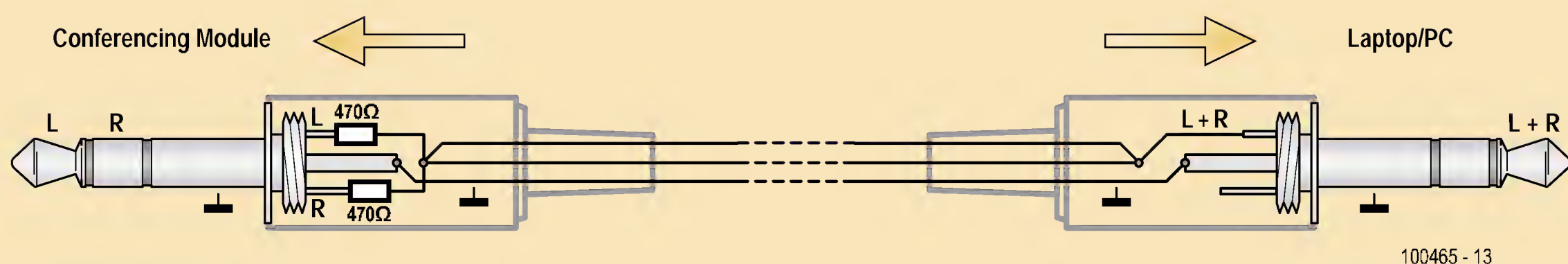
Figure 3. The 'Skype cable' simply sums the left and right channels.

the distance between the membranes becomes greater than half a wavelength, then the gain drops. When, for example, the membranes are mounted with a separation of 2 cm from each other, the corner frequency amounts to:

$$f = \frac{v}{\lambda} = \frac{330\,m/s}{0.04\,m} = 8.25\,kHz$$

where v = speed of sound in air. This is still sufficiently high for speech.

To ensure the simplicity of the design we decided, as a first instance, to allow for a maximum of two identical modules to be connected. One of the modules is connected with a mini jack plug to the other module and the latter is then connected with a mini jack to the Line (audio) input of the PC or laptop. Now it would of course be very useful if the gain of the circuit could be adjusted. For this purpose a simple potentiometer has been added to the circuit. And a Mute button — for when you would like to discuss something amongst yourselves without disturbing the entire meeting — is also an essential feature.

How does it work?

The starting point for the design of this circuit was to make a simple schematic that would realise the desired functionality. And simple it certainly is, see **Figure 2**. Only a single opamp and a handful of resistors, capacitors and connections. Nevertheless we will talk you through its operation.

The microphone capsules are connected via PCB headers. In this way you can easily mount the capsules in the correct place inside an enclosure. In the sketch of **Figure 1** you can see how we mounted the capsules (there is another one on the opposite side at the same height) as high is possible in a triangular enclosure. The closer together they are, the better. To provide the FETs in the capsules with their power supply voltage, the outputs of the capsules are connected with R1, R2 and P1 to the power supply. The values have been selected so that the input of the opamp is at about half of the power supply voltage. This is not that important for the output signal of MIC1, because it is DC decoupled using C2 from the input of the amplifier. This is very important for the output signal from MIC2, because it determines the DC bias of the entire circuit. P1 has to be adjusted so that the voltage at

the input of the opamp is equal to half the power supply voltage, i.e. 4.5 V.
Resistors R3 through R6 and pot P2 determine the gain of the opamp. An amplification A of up to 10 times was selected, which turned out to be sufficient in practice. The output voltage is expressed by:

$$v_{out} = -v_1\left(\frac{R6}{R3}\right) + v_2\left(\frac{R5+P2}{R4+R5+P2}\right)\left(1+\frac{R6}{R3}\right)$$

where v_1 and v_2 represent the voltages at the inputs of the opamp.

As can be seen from the formula, P2 has an effect on the gain that is applied to the signal from MIC2. This allows for differences between the two capsules to be eliminated and also allows for the characteristic of the microphone formed by the two capsules to be changed from cardioid or hyper-cardioid to figure-of-8 pickup pattern. If the modules are exactly the same then P2+R5 would be exactly equal to R6 and there would be a perfect figure-of-8 characteristic.
Capacitor C4 prevents the tendency of the opamp to oscillate and R7 protects the output from short circuits. C5 decouples the

volume control potentiometer from the DC voltage that is present on the output. To make it easy to select a location for the volume potentiometer (P3) in the enclosure this potentiometer is also connected to a header. The Mute button can them simply be connected in series with this potentiometer.

To allow the modules to be connected together, K1 and K2 are connected in such a way that the output signal of a module is always connected to the tip of the jack plug and the looped-through signal from the other module is on the ring of the jack plug. In this way it does not matter how the modules are connected together and the signals from the modules are always on the left and right channels of the output connector.
To keep the number of wires on the meeting table to a minimum we chose to power the circuit from a 9-V battery. The current consumption of the circuit is only about 2 mA, so a 9-V battery with a capacity of 250 mAh will last about 125 hours.

How do you build it?
A small printed circuit board was designed for this circuit (see **Figure 4**). The design is available free in PDF and Eagle file format from the Elektor website [3]. The assembly of this is a piece of cake. The headers K3 through K7 are optional; the wires can, of course, be soldered directly into the circuit board.

The mechanical construction of this project does require some attention however. In particular the building-in of the microphones can be a bit tricky. These need to be attached in a 'decoupled' manner, otherwise they will transmit direct vibration. In addition to the 'normal' sound transmission through the air, every tap on the table will also be amplified by direct mechanical coupling, which has a considerably negative influence on the sound quality and intelligibility. A good solution is to wrap the capsules in foam rubber and then glue this into the enclosure. The triangular enclosure suggested in **Figure 1** is an idea of ours, but is not something that we have worked out the details for. Our prototypes consisted of two wooden boards, which were attached

COMPONENT LIST

Resistors
R1 = 10kΩ
R2 = 22kΩ
R3,R4 = 33kΩ
R5 = 220kΩ
R6 = 330kΩ
R7 = 220Ω
P1 = 22kΩ trimpot, horizontal
P2 = 250kΩ trimpot, horizontal
P3 = 2.2kΩ logarithmic potentiometer

Capacitors
C1 = 10µF
C2 = 100nF
C3 = 22µF
C4 = 100pF
C5,C6 = 1µF

Semiconductors
IC1 = TL071

Miscellaneous
K1,K2 = 3.5mm jack socket, PCB mount
K3–K6 = 2-pin pinheader
K7 = 3-pin pinheader
Mating headers for K3–K7
S1 = pushbutton, 1 make contact
S2 = on/off switch
Electret microphone capsule, e.g Panasonic
    MCE-2020U or WM-61A
PCB # 100465-1, see www.elektor.com
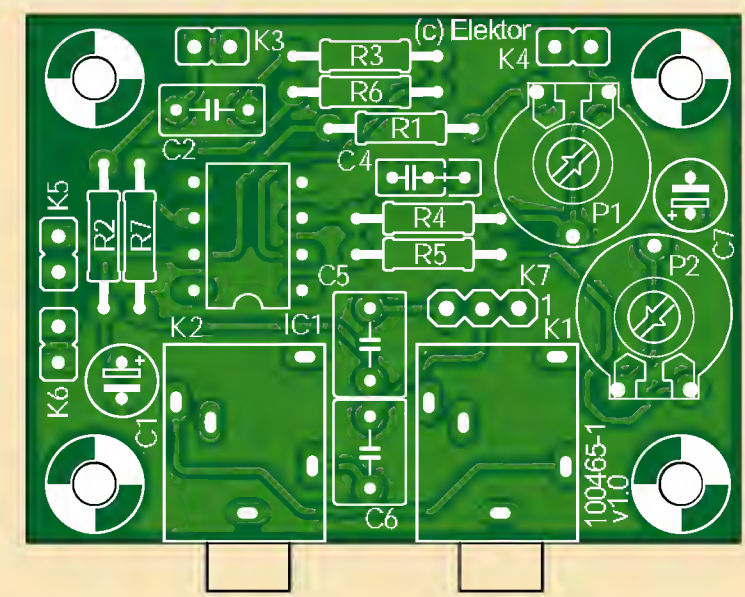
Figure 4. With a simple circuit comes a simple printed circuit board.
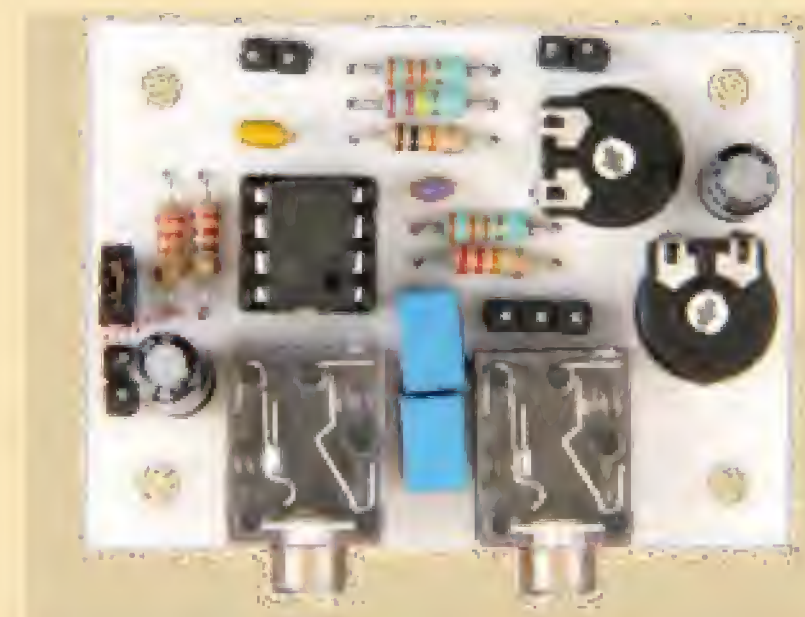
to each other to form an inverted T, where the microphone capsules were attached to the vertical board in a piece of foam rubber.

Connecting and adjusting
Adjust the output voltage of the opamp with the aid of P1 to half of the power supply voltage (4.5 V). With P2 the pickup pattern of the combined capsules can be varied from cardioid or hyper-cardioid to figure-of-8. Figure-8 is the best for situations such as that in **Figure 1**, a cardioid or hyper-cardioid is better when the sound has to be picked up from only one side of the module. Connect one module to the other via K1 of one and K2 of the other. Now the module that has K2 still available is connected to the line-input of the PC or laptop. Don't forget to set the software that you are using for your online meeting to use the line-input as the source of the sound.

Incidentally, when testing with the commonly used Skype software, it turned out that this program does not sum the audio line inputs but only uses the left channel, while other programs such as Windows Live Messenger do properly sum the left and right signals together. With our circuit it would then only transmit the sound from one module only. To solve this we have made a 'skype-cable' (see sidebar).

(100465-I)

Internet Links
[1]   http://en.wikipedia.org/wiki/Microphone

[2]   www.elektor.coml/010123

[3]   www.elektor.com/100465

Perfectly Balanced
Noise-free volume control for Hifi amplifiers

JBy an Breemer (The Netherlands)

Decent potentiometers for setting the volume in amplifiers are still quite expensive. However, there is a cheaper alternative, and it's better too! The PGA2311 made by Texas Instruments can be used to make an almost perfect volume control. A simple circuit using an 8-pin Freescale microcontroller and a linear potentiometer are all you need to control this IC.
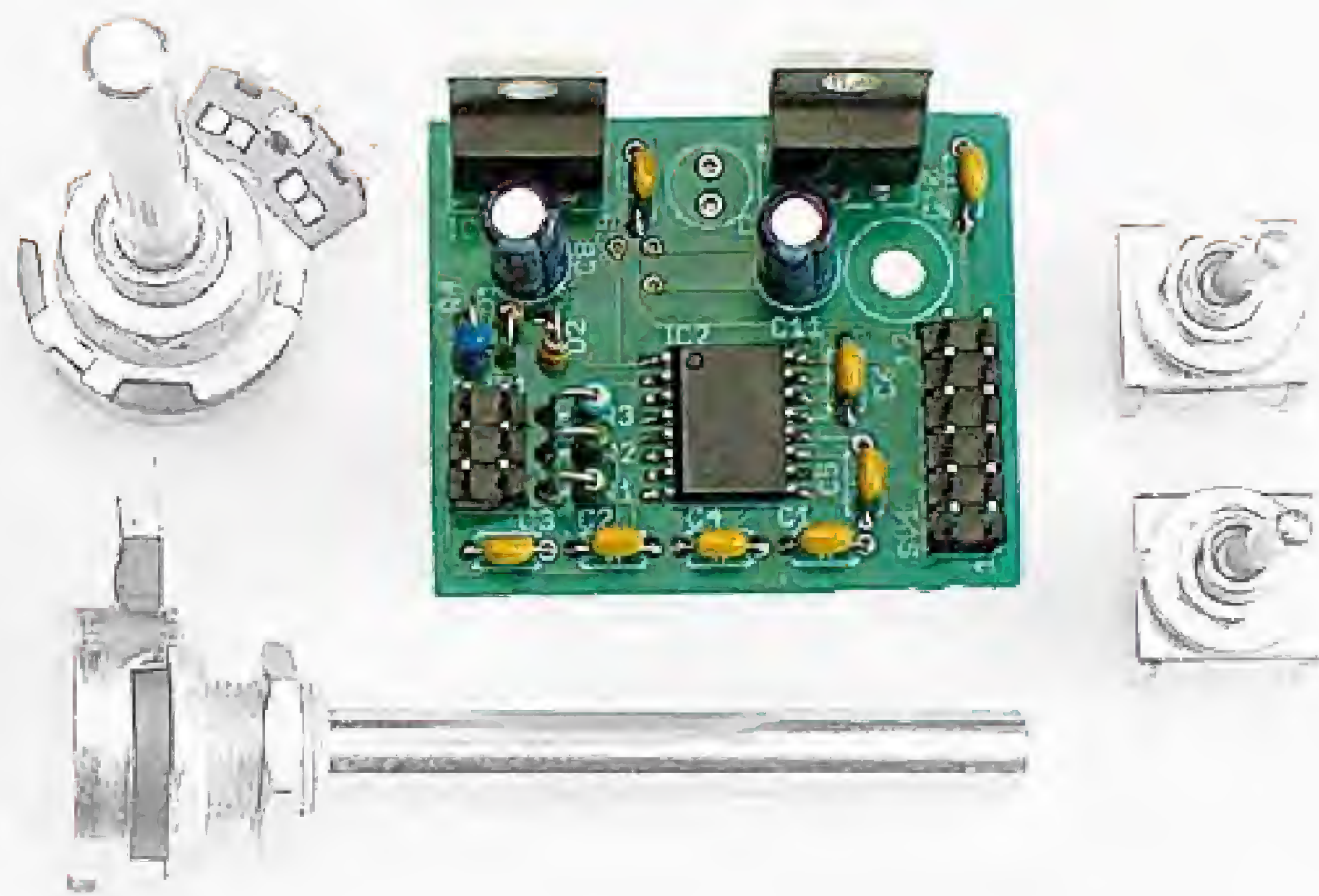
The volume control in audio amplifiers is usually implemented using a (stereo) potentiometer, which in reality is just a resistor with a variable tap. This tap can be adjusted with the control knob, setting the volume to the required level. The potentiometer is often a logarithmic type. This means that for the same angle of rotation anywhere along its track the corresponding change in dBs will be the same (with the resistance changing logarithmically with respect to the angle of rotation).

Even high quality amplifiers occasionally suffer from problems caused by potentiometers. They sometimes exhibit a number of failings:

- The wiper can make a bad contact with the resistance track, resulting in a crackling sound when the knob is turned.
- Under certain circumstances a bad wiper contact can result in non-linear distortion (creation of harmonics and sum and difference frequencies).
- The tracking of stereo potentiometers leaves a lot to be desired. Even the most expensive types are only slightly better than the cheaper ones.
- Volume controls suitable for more channels (5.1, 7.1) are very difficult to get hold of.

Alternatives

We can think of several alternatives for the potentiometer:

- An expensive potentiometer with very high specifications. As we mentioned earlier, these still aren't perfect and they can easily cost tens of pounds.
- A stepped volume control with a large number of steps and resistors. Even if you can find them, they'll be very expensive (over £ 70). It would be a good alternative for home construction, but the problem is that you need a switch with a large number of steps. It's possible to buy a 24-way switch, but that isn't enough for a smooth volume change. You really need at least 40 steps and such switches are very difficult to find and rather expensive.
- A circuit using light dependent resistors (LDRs). These will also have tracking problems.
- An analogue circuit, whether integrated or not. It's possible to make an analogue multiplier circuit or get one in chip form. The problem with these circuits is that the distortion is often too large and the range too small.
- Digital potentiometers. Intersil [1] has a number of these in its catalogue, but these are mostly linear. It's also difficult to guarantee that two mono potentiometers will exhibit good tracking.
- A digital integrated circuit, specifically designed for audio use.

In this article we'll cover a circuit that comes in the last category.

The circuit

In the April and May 2004 issues of *Elektor Electronics* the design for the High-End-Preamp was published. In the preamp, the volume control was implemented using a PGA2311AP IC made by Texas Instruments. The specifications for this IC are superb; there simply isn't any mechanical potentiometer that can improve on it.

The required digital control signals are fairly complex, so a microcontroller was added to take care of those. The SpYder project from Freescale and Elektor (published in the March 2007 issue) is just perfect for this task. Using this, it becomes easy to control the PGA.

The PGA2311 is controlled on the digital side by the /CS, SDI and SCLK signals, originating from the microcontroller (see **Figure 1**). On the analogue side are the analogue inputs and outputs, made available on a dual-in-line header, JP2. The linear potentiometer is connected via 6-pin

Specifications

- Stereo volume control with steps of 0.5 dB over a range of 128 dB
  (optional; a smaller range was chosen for this project)
- Extremely low distortion
- Extremely good tracking
- Controlled via a standard linear rotary or slide potentiometer
- No interference caused by the digital circuit; intelligent software turns off the external digital
  signals about a second after the last change in volume
- Optional mute switch and activity LED
- Can be added to virtually any amplifier
- No exotic or expensive components

header JP1A. Via this connection you can also have a mute switch and an optional LED that indicates when the control IC is updated by the microcontroller (see **Figure 2**). The voltage returned by the potentiometer is filtered by a capacitor, before being fed to the A/D converter of the microcontroller. This reduces the noise in the signal, which is important as it prevents the output of the A/D converter from flicking backwards and forwards between two states when the potentiometer is exactly halfway between two bits.

The PGA2311 has a fairly low input impedance (about 10 kΩ). It is recommended to use a source with a low output impedance in order to keep the distortion to a minimum. This shouldn't be a problem with modern audio sources such as CD/DVD/SACD/MP3 players, receivers, tuners, satellite receivers and PCs. These devices usually have an output impedance of 500 Ω or less. Older equipment (in particular those with DIN sockets) may have a higher output impedance. In that case the PGA2311 has to be preceded by a (simple) buffer stage.

The output impedance of the PGA2311 is very low. It won't have any problems driving all types of amplifier.

The PGA2311 can deal with input signals of almost ±5 V maximum, which is 3.5 V RMS. Larger signals are suddenly distorted a great deal. The output signal can go as high as 4 volts peak-to-peak, or 2 V RMS.

Power supply

There are several ways in which the supply voltage can be generated, depending on the amplifier to which the volume control is added. We have taken account of this in the design of the circuit. Table 1 has an overview of several possibilities. This overview is far from complete, since there are many other possibilities and every amplifier is different.

The circuit requires a supply of +5 and -5 Volts. The current consumption from the positive supply line during the changing of the volume is about 20 mA and 15 mA when the circuit is at rest. For the negative supply line this figure is about 15 mA at all times. The microprocessor operates at 3.3 Volts. This voltage is derived from the +5 volt supply using diodes D1 and D2. This isn't

exactly according to the worst-case specifications, but from experience we found that it still works reliably.

Should the audio outputs be loaded with, for example, 600 ohms from a set of headphones, it can easily add another 10 mA to the current consumption, depending on the output volume. In that case the symmetrical 5 V supply lines need extra buffering with electrolytic capacitors of a few hundred microfarads at 6.3 V. It is also recommended to use the TO220 versions for the voltage regulators.

Construction

The author has made several kits of parts available for this project via his website [2]. This also includes variations for the different power supply options.

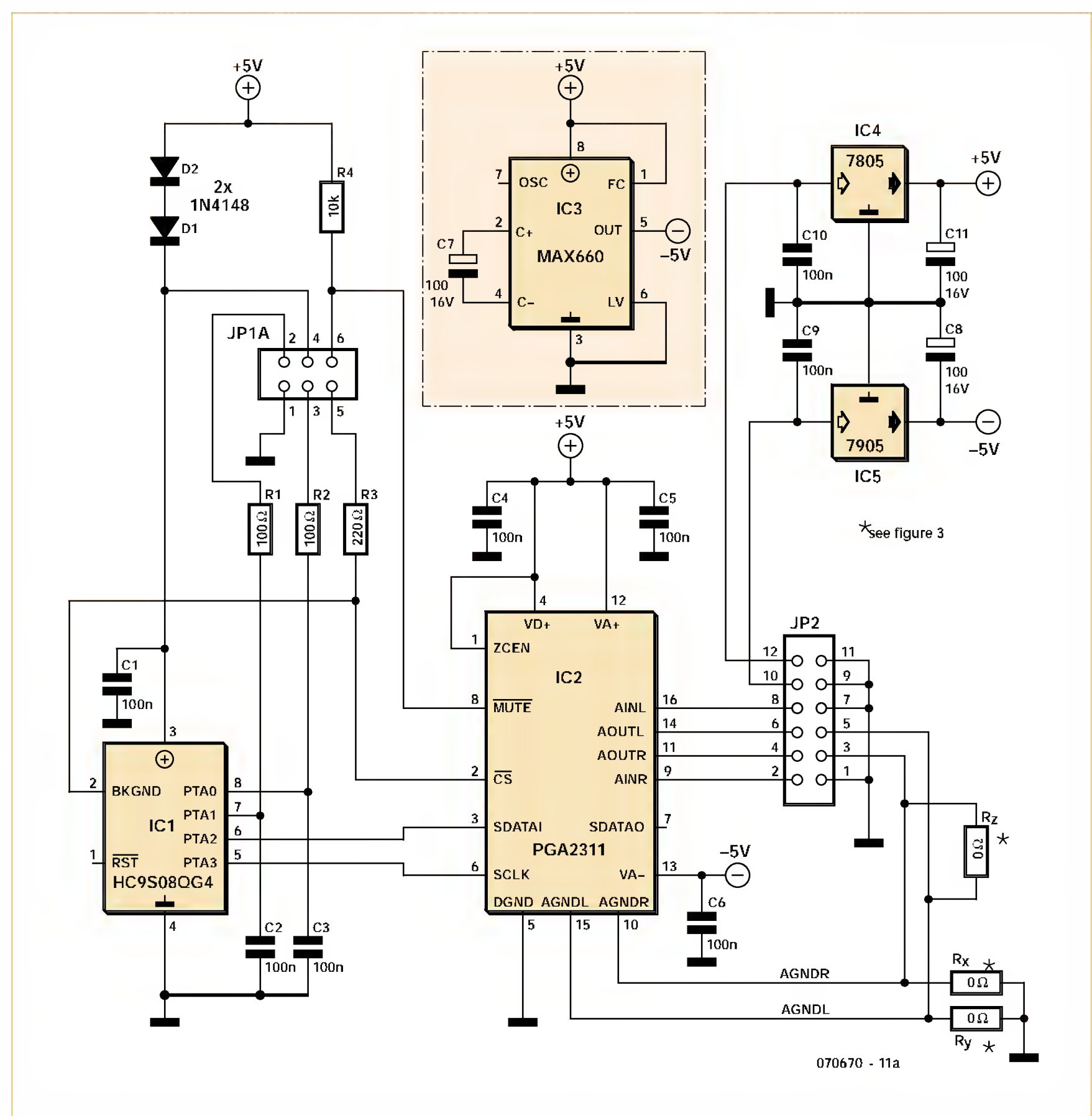There is a choice from a complete kit of

Figure 1. The circuit diagram for the volume control is fairly straightforward. In the coloured section is the optional MAX660, which is fitted depending on the type of power supply used.
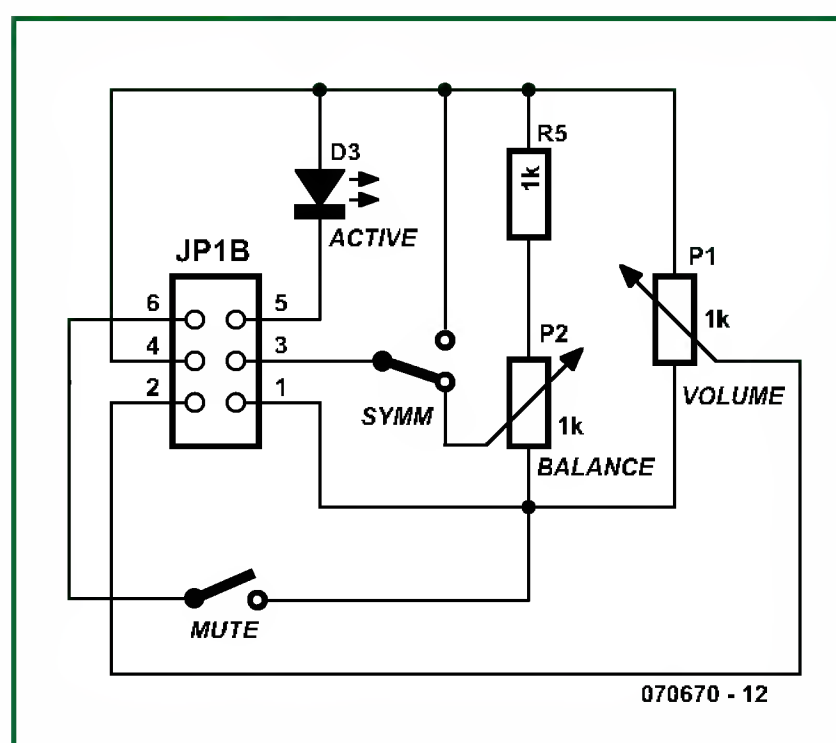
Figure 2. The potentiometers, switches and indicator LED aren't mounted on the PCB. They could, for example, be mounted on the front panel of the amplifier.

Figure 3. Suggested ground plane layout.

parts, or the PCB and a ready-programmed controller. But it is also possible to order just the PCB and have somebody program the controller for you, or even do it yourself. In that case you'll need to find the remainder of the components yourself.

During the construction of the circuit it is best to start with the smallest (SMD) components, as usual. IC1 comes in an SOIC-8 package, which makes it a prime candidate for the hot tin. Then it's a matter of common sense and keeping the power supply option in mind. JP1A should be connected to JP1B using a one-to-one ribbon cable. It's possible to leave out JP1B and solder the wires directly to the potentiometers and switches. Make sure that you don't forget about R5. There is no space reserved for this on the PCB and it needs to be soldered directly to the potentiometer.

The input and output signals are connected to the circuit via JP2. The inputs are on pins 8 (left) and 2 (right), the outputs on pins 14 (left) and 11 (right). The supply is connected to pins 10, 12 and any of 1, 3, 5, 7, 9 or 11. Refer to the circuit diagram in Figure 1 for more details.

The Software

The starting point for the software is the 'Blinking LED' demo program in the Elektor/Freescale SpYder Discovery kit, which is then modified for our use. Only 'main.c' needs to be changed, with the remaining files staying the same. The inset 'DIY programming' describes the steps you have to take to program the firmware into the microcontroller yourself. There are four routines inside the main.c program:

1. void PeriphInit (void)
Compared to the 'Blinking LED' demo, the only change is the data direction of the out-

put ports. Pins 5 to 8 have now been configured as outputs. The configuration for the analogue input (pin 8) is overwritten later by the A/D converter configuration.

2. void SendData (VolumeSetting)
This routine is written in assembly language and controls the /CS, SDI and SCLK lines that go to the PGA2311. There is a sub-routine that is called twice, once for the right channel and once for the left channel, both with the same data (unless the balance control is used, when the volume for the right and left channel can differ). A couple of NOP (No OPeration) instructions have been added to make the run-time identical for both branches. This isn't vital for the operation of the program, but it makes the interface signals on an oscilloscope look a lot better. The /CS line stays low at the end so that the activity LED stays lit. It's only when no new data has to be sent that the /CS line goes high.

3. Byte AdConvert (void)
This configures the A/D converter and starts a conversion. The routine then waits for the conversion to finish and finally returns the result. It isn't really necessary to configure the converter every time, but it doesn't do any harm either. The converter is used in a 'slow' 8-bit mode, but that is still more than fast enough for this application.

4. main (void)
This is the starting point and also the 'body' of the program. The 'main' loop never ends. After calling the initialisation routine *Periph-Init()* once, there is an infinite loop where 100 A/D conversions take place. The result is averaged out and compared with the previous result. If there is a sufficient difference the new result is scaled and sent to the

PGA2311. This process happens about 100 times per second, but stops after about a second during which no changes took place. That way there won't be any digital interference in the audio signal.

The criterion for 'no changes' is somewhat loosely implemented. The change has to be at least three units before the sending process is started again. This stops a change being made when the potentiometer is positioned exactly between two A/D conversion points. It would of course be quite noisy if the volume was continuously changed up and down by a 1/2 or 1 dB step. Incidentally, a little trick is used for the balance control. As soon as the microcontroller detects that the balance potentiometer has moved beyond ¾ of its range, it is interpreted as 'balance control is turned off'. The left and right channels are then amplified or attenuated by the same amount. The 'SYMM' switch (**Figure 2**) has the same effect, by the way.

All of the HEX file and source code files can be downloaded free of charge from the Elektor website [2].

Interference suppression

There are three ways in which audible interference could be introduced:

- Interference from the microprocessor signals finds its way into the audio signal path.
- When the gain is changed it can introduce audible clicks in the audio signal, because the change is so abrupt.
- The result of the A/D conversion could vary slightly, even though the state of the potentiometer hasn't changed.

The first type of interference is all but eliminated because all external microcontroller signals are shut down about one second after the volume control stops changing. The only activity on the chip itself is the reading of the A/D converter.

The PGA2311 has a facility where the gain is only changed when the audio signal passes through a zero-crossing point. This is set with pin 1, ZCEN, and works very well. It is only when music has dominant bass sounds of 50 Hz or less that you can hear an occasional soft click when changing the volume. This therefore mostly suppresses the second source of interference.

The final type of interference, where the A/D conversion varies, would cause the gain to change by ±0.5 dB. This is can be heard clearly, not as a change in volume, but as a soft click when the volume is changed. This can be trapped in the software. About a second after the potentiometer stops changing the PGA2311 stops updating the volume control. The /CS signal goes high and the LED turns off. Should the LED keep flashing you have probably used a very bad quality potentiometer for P1 and it should be replaced.

The right range

The PGA2311 has a control range from -95.5 dB to +31.5 dB. This is impractically large. In many cases a gain of 31.5 dB (that is about a 30x gain in the voltage) isn't required. Neither is an attenuation of -95 dB necessary. It was found that settings of 0 dB to -60 dB were sufficient. That's as quiet as a whisper when 0 dB corresponds to about 100 dB(A). There is a facility in the software to set the range, using the constants 'UpperLimit' and 'LowerLimit'. The standard setting is a maximum gain of 0 dB (1, or 'unity gain') and a minimum gain of -60 dB (1/1000). When the potentiometer is turned right down the control IC goes into a soft-mute mode and the audio signal is muted.

Listening tests

After building the circuit into his preamplifier, the first impressions of the author were so good that he wondered if the circuit was active. When changing the volume a soft click was sometimes heard if the music was dominated by strong, low bass sounds, such as occur in some organ music. As soon as the volume control was let go, this stopped. The range was originally set between -80 dB and 0 dB. This easily covers a range from deafeningly loud down to a quiet whisper. After several months of use it turned out that the range was too large; the quietest third of the control was never used. A few experiments showed that a range of 0 dB to -60 dB was the best.

(070670-I)

Internet Links

[1] www.intersil.com

[2] www.elektor.com/070670

DIY programming

It's of course also possible to program the microcontroller yourself, instead of buying a ready-programmed controller from the author. The instructions below show how the firmware can be easily programmed, using the "SpYder Discovery kit for Freescale MC9RS08KA, MC9S0QD and MC-9S08QG Microcontrollers" many of you may have as result of special promotion by Elektor in 2007.

- The SpYder kit contains a USB stick with a microprocessor and a CD-ROM. Install the software according to the instructions given in the kit.
- Try out the demo of the Flashing Light LED according to the instructions given in the kit. If all is well you know that everything has been installed properly and works as it should.
- Make a copy of that project (...\USBSPYDER08\QG4\Demo) in another folder. But take care! When you make a copy of a project, some absolute paths remain that point to the old folder. You should therefore edit all references to the old path such that they point to the new folder, otherwise you'll include parts of the old project.
- Check once more that everything still works (verifying that the paths are correct).
- In the sub-folder ...\Sources\ replace the file main.c with the new main.c file, which should be downloaded from the Elektor website.
- Start the development environment by double-clicking on the file "Demo.mcp" in the new sub-folder.
- Compile the program (F5) and load it into the microprocessor in the stick ('Enter' in the debugger).
- Close the development environment, remove the stick from the USB socket and put the programmed microprocessor onto the volume control PCB.
- Check that everything works. While the potentiometer is turned the LED should stay on. About one second after you stop turning the LED should go off.

Table 1. Power supply options

Option	Available voltages	Mount
1	12 to 15 volts symmetrical	IC4 and IC5 in low-power case. Do not mount IC3.
2	15 to 30 volts symmetrical	IC4 and IC5 in TO220 case. Do not mount IC3.
3	12 to 30 volts asymmetrical	IC4 in TO220 case and IC3. Do not mount IC5.
4	Over 30 volts, pos or neg	As 1, 2, or 3, but the voltage first needs to be reduced to a value the regulators can cope with.
5	None	A home-made supply: a transformer with 2x9 to 12 V, a bridge rectifier and 2 electrolytic capacitors of about 500 μF, 25V. Then choose either option 1 or 2.

N.B.: The options where IC3 is used do have a small disadvantage: there is a continuous square-wave of about 40 kHz with an amplitude of 5 Vpp across C7. In combination with a CD player that doesn't adequately suppress the 44.1 kHz sampling frequency it could give rise to intermodulation signals.

COMPONENT LIST

Resistors
R1,R2 = 100Ω
R3 = 220Ω
R4 = 10kΩ
R5 = 1kΩ
P1,P2 = 1kΩ linear

Capacitors
C1-C6,C9,C10 = 100nF
C7,C8,C11 = 100µF 16V

Semiconductors
D1,D2 = 1N4148
D3 = LED, free choice

IC1 = HC9S08QG4CDNE (Freescale)*
IC2 = PGA2311 (Texas Instruments)
IC3 = MAX660 (Maxim)**
IC4 = 78x05**
IC5 = 79x05**
* available ready-programmed via
   www.breem.nl
** whichever applies

Miscellaneous
pinheader 2x3
pinheader 2x6
flatcable 2x3
flatcable 2x6
mute switch, single on/off
symmetry switch, single pole on/off
PCB available via www.breem.nl

Elektor Proton Robot
A versatile platform for learning and experimenting

By Bart Huyskens (Belgium)

The Proton robot from Elektor is a versatile platform that's suitable for students, enthusiasts and professionals alike. The robot can operate with a variety of microcontroller families, and it supports a broad spectrum of sensors and actuators. This is an ideal opportunity to invest in a unique combination of knowledge and fun.

The new Proton robot described in this article is a DIY system that can easily be extended in all sorts of ways. You can purchase the robot as a complete kit or ready-made, or you can purchase the individual components separately. This makes it easy to adapt the system to your own wishes and needs, as well as the available budget. **Figure 1** shows an overview of the Proton robot and its key features.

Your choice of microcontroller
The most important feature is undoubtedly the fact that you, as the user, can **personally** decide which microcontroller to use to control the robot and which language and/or programmer you want to use for this purpose. When you buy the robot, you can choose from an add-on PCB for the PIC15F887 or a PCB for the AVR ATmega32, but with a piece of prototyping board and a minimum of soldering work you can also control the robot with virtually any microcontroller, as long as it supports a bit of digital I/O, a couple of analogue inputs, I²C and RS232. Experience with I²C, RS232 or analogue inputs is by no means necessary. The robot comes with macros that radically simplify the task of configuring the necessary settings. As you can see from the overview in Figure 1 and the block diagram in **Figure 2**, most of the components are controlled over the I²C bus. The I²C protocol is very simple, and most microcontrollers have a hardware I²C interface module on board.

Your choice of programming language
Example programs in both Flowcode and C have been developed for all of the modules of the Proton robot, for both the PIC and the AVR microcontrollers. The C compiler for the PIC MCU is Hitech-C Lite, and for the AVR MCU it is WinAVR GCC. Both of these tools are free, but
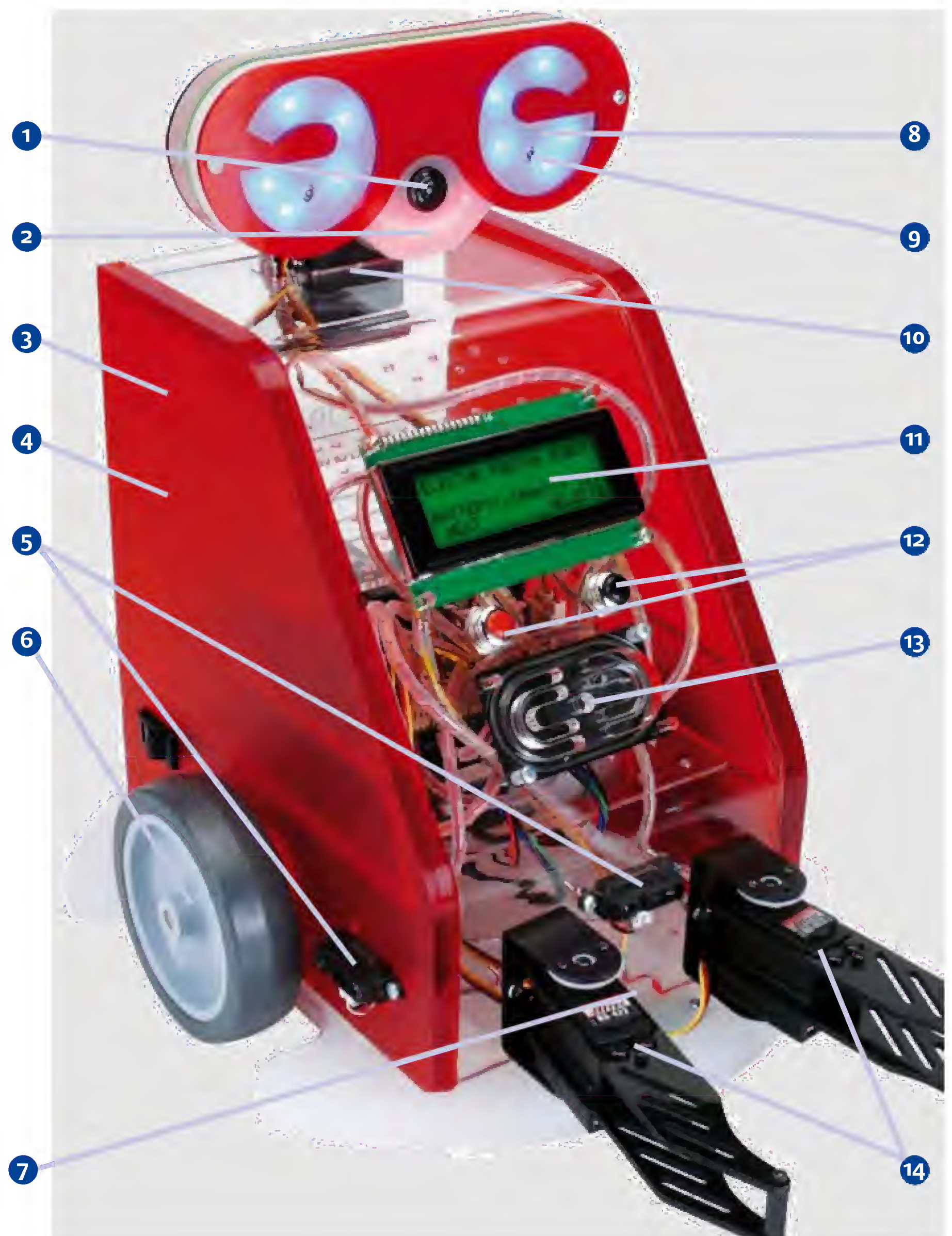
you can also select other compilers and languages. The developer has created a customer header file for both of these C compilers, which makes programming in C a lot easier. For Flowcode there is a set of macros that simplify the relatively complex settings. In both cases, all of the code is open source, so there is nothing to stop you from adapting it to your own purposes.

The developer used the AVRISPM Mk2 and PickIT2 programmers, both of which are good low-cost programmers available from Atmel and Microchip. However, other types of programmer hardware can also be used.

Extensive support

A variety of activities are being organised around the Proton robot to support its users.

User guide: First of all, there is a very extensive user guide that provides useful information for everyone from neophytes to profes-

sionals to help them program the various components of the robot. Every component is described in detail, and for each component there are four fully coded examples that show how it can be programmed in Flowcode or in C: FC-PIC, C-PIC, FC-AVR and C-AVR.

Forum and website: A separate page on the Elektor website will be totally dedicated to the Proton robot. There you will always find the latest version of the user guide, various tips and tricks, and of course downloads. On the Elektor forum you'll be able to share your ideas and experience with other Proton users and the Proton developer. You can also share code, schematics and videos on this page.

Creative challenge: We challenge you to develop your own add-on projects and connect them to the robot. The motherboard is equipped with spare I/O, A/D, RS232, SPI and I²C ports, and there is plenty of room on the robot to install your own projects. Some ideas: control the robot with your mobile phone, via Bluetooth or

	Features	
1	Ultrasonic distance sensor	An ultrasonic distance sensor controlled by simple I²C commands. The readings are provided in centimetres, inches or microseconds. **Function:** Measuring distances from 4 to 300 cm.
2	8 LEDs in the mouth	The robot's mouth is formed by eight LEDs that can be driven individually in normal mode. In audio mode they serve as a VU meter for the music from the audio module, so the robot's mouth moves in sync with the speech or music. **Function:** Indicating the robot's response to sound perception.
3	Piezoelectric speaker	A small high-impedance speaker connected to a digital output. It can generate all audible frequencies. **Function:** Beep sounds, alarms, ringtones.
4	8 LEDs in the body	Connected to eight digital outputs of the MCU. **Function:** Debugging and initial program steps.
5	3 infrared distance sensors	Three analogue IR sensors that can measure distances from 4 to 80 cm. **Function:** Detecting and/or avoiding obstacles.
6.	Motor drive module	This robust DC motor drive module is controlled over the I²C bus. Rotary encoders in the motors provide constant feedback on the position and angular motion of the motor module. These values can be queried at any time by the MCU, along with the battery voltage and the current drawn by each motor. **Function:** Setting the speed and rotation direction of the two motors with the aid of simple commands.
7	3 line detectors	Three infrared sensors underneath the robot, connected to three digital inputs of the MCU. **Function:** Following black or white lines.
8	LED eyes	Five RGB LEDs in each eye, individually dimmable under PWM control in order to create any desired colour. **Function:** Generating light effects.
9	2 phototransistors	Two phototransistors in the head, which communicate over the I²C bus. **Function:** Measuring light intensity and determining the location of a light source.
10	2 servomotors	These two motors move the robot's head under the control of I²C commands. **Function:** Adjusting the vertical and horizontal position of the head.
11	LCD	Messages appear on a large blue LCD with four lines of twenty characters each. The included example programs and macros make driving the LCD very easy. **Function:** Displaying measured sensor values or user interface menus.
12	Red and black buttons	Two buttons connected to two digital inputs of the MCU. **Function:** Selecting menu items.
13	Audio module	This module plays MP3 and WAV files directly from an SD card. It is controlled by a set of RS232 commands. **Function:** Playing MP3, WAV and other audio files from an SC card (to enable the robot to speak, among other purposes); saving data and readings on an SD card; generating equaliser data.
14	Gripper	Optional gripper. **Function:** Picking up and moving cans and other small objects.

via WiFi; let robots communicate with each other; fit a camera and experiment with image recognition; let your robot respond to voice commands; etc. — the sky's the limit. Document your add-on project, and maybe it will be selected for publication in Elektor or nominated for an attractive prize at one of the events organised in connection with the Proton robot. Among other activities at these events, we plan to hold contests with a special challenge on each occasion.

Kit or fully assembled

You can order the robot as complete kit for DIY assembly, or you can order the individual components separately. A detailed assembly guide with photos and instructions helps you put them together. All circuit boards are fully assembled and tested; all you have to do is mount or fit them and solder a few cables. This is certainly not difficult, although a bit of experience with soldering will come in handy. If you wish, you can also order the Proton fully assembled and equipped with extensive test software. See the Elektor website for all of the options!

(110263-I)
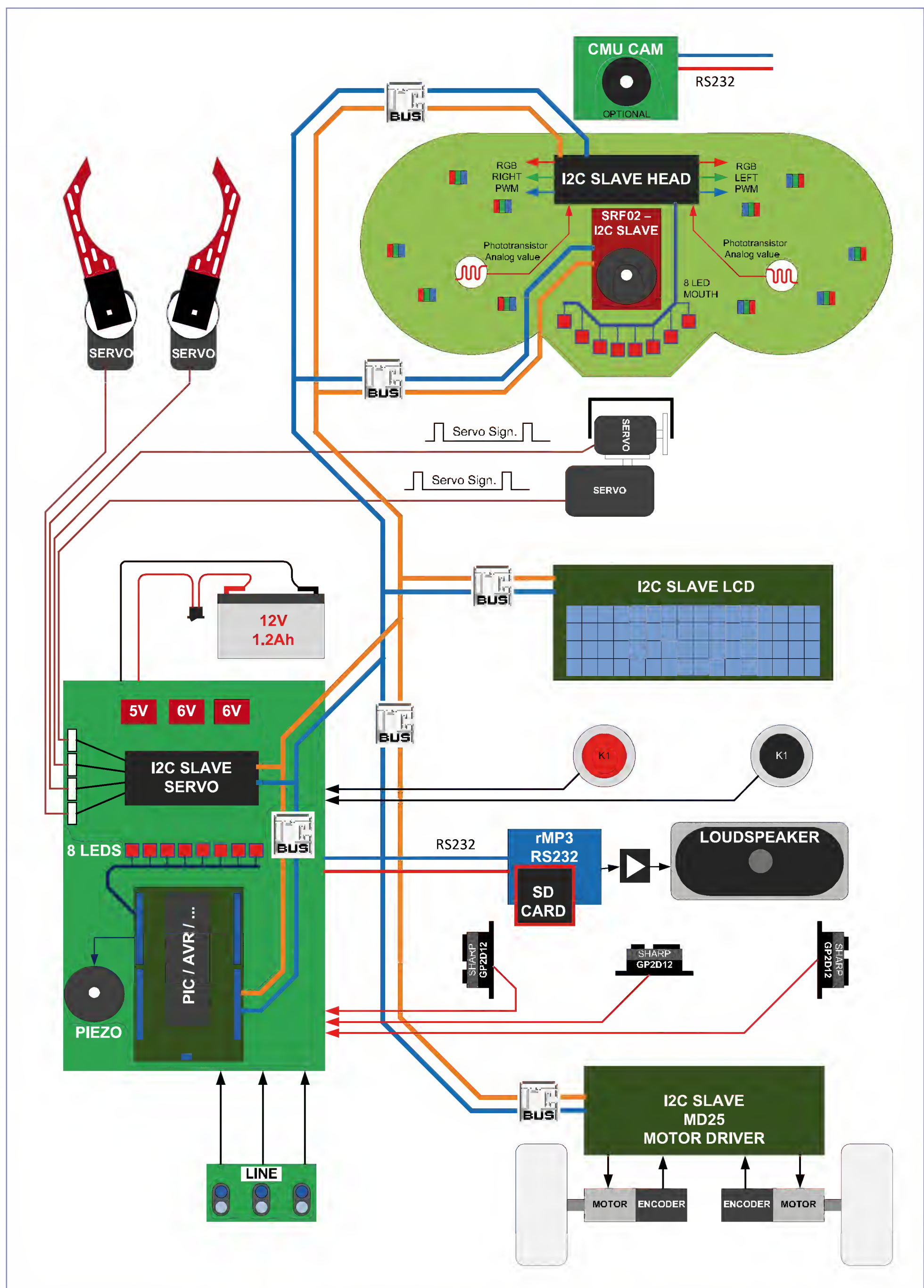
Further information and ordering options

www.elektor.com/proton

YouTube video

http://www.youtube.com/watch?v=4WiH3LCzYJk

About the author

The Elektor Proton robot was designed by Bart Huyskens. Bart is an instructor of Electronics and Embedded Systems at the St-Josef Institute in Schoten (Belgium) with a passion for his work, and he has already designed several robots, including the popular Formula Flowcode robot and the Robu robot. He spent over two years developing the Proton, and the results speak for themselves.

Help! I'm stuck...

By Thijs Beckers (Elektor Labs)

In the lab we don't just design circuits. Technical questions also find their way to the heart of the company. Despite the fact that we do our utmost to test the published designs thoroughly, write the articles as clearly as possible and make as much supplemental information as possible available via our website, it can of course happen that something goes wrong. Not surprisingly, a flood of questions then comes in. Most of these have been sent using the Contact Form on the Elektor website [1].

A 'good' example of this is what happened to the Digital Multi-Effects Unit from the September 2010 issue. In the rush to meet the deadline for the article some of the electrolytic capacitors were drawn the wrong way round in the circuit diagram (C23, C25, C30 and C32), with the result that they also had the wrong polarity on the printed circuit board. In principle this wasn't such a big problem; since the voltages across the electrolytics were relatively low they shouldn't be affected by the polarity reversal. A much bigger problem was that the supplier made a mistake in programming the first batch of the ATmega8 on the main board. Several fuse settings were wrong, which made the chip use its internal clock, whereas the intention was that it should use the external 8 MHz crystal. The result: just a flashing cursor appeared on the display. This led to a number of disap-

"The amplifier I built in 1975 from your mag is becoming noisy. Could you please send me an update and a new PCB?"

pointed customers and a large amount of extra work for the lab, customer services and the logistics department. We obviously try to correct mistakes such as these as soon as possible after they are discovered. All got rectified in the end.

Many other technical questions relate to much simpler subjects, such as: "Where should I put capacitor C15?" and "Which type of solder should I use?". These types of question are of course a piece of cake for our colleague Jan Visser. But some questions made him struggle: "So what's the difference between an ATmega and an 8051 core?" "I would like to use circuit *xyz* from Elektor, but I want (*xx*+33) outputs instead, the supply voltage should be 12 V and I'd also like to have 512 KB memory in the processor... What changes do you suggest?" and "I'm looking for a particular sensor to measure ... Could you find a suitable type for me that isn't too expensive and preferably is available around the corner?". We hope you'll understand that we don't have enough time to research the answers to such specific questions. The difference between an ATmega and an 8051 processor can't be explained briefly without having an

in-depth knowledge of both processor families. Furthermore, there is a wealth of information available for both processors, which should help you choose the one that is most suitable for your project.

A question that comes up regularly concerns the availability of components. SMD parts in particular are rarely in stock in electronics stores. In that case you could try asking the shop if they can order some of the components for you. It is of course not possible to turn up with an Elektor parts list and ask them if they could order the lot from, say, Farnell. Some dedicated components are usually available as one-offs; 'standard' components are often only available in multiples of 10, 25 or even 100. This is not such a big prob-

"Could you please design a LED message board for the restored windmill in our village, similar to your LED Spin-Top?"

lem with SMD resistors, which cost only a few pence each, but when it concerns capacitors that cost £0.25 each, and you have to order a minimum of 10 and you have to do this for five different values, you'll soon see the cost mount up. It can therefore take some time to figure out how best to deal with this. In some countries people club together to buy bulk quantities of components, generally via forums that are visited by hobbyists, i.e. people who also ran into the same problems. As far as availability is concerned: here at Elektor we do our best to choose as many current components as possible. However, we can't prevent that some components are declared obsolete by the manufacturer after only three years (and sometimes sooner). In the electronics world changes happen almost as fast as in the computer world: these days a ten year old circuit is almost considered prehistoric, and dedicated ICs in particular are often difficult to find...

To help us deal with questions as quickly and thoroughly as possible we have added a number of input boxes on our online query form [1]. It's a great help if you take the trouble to enter the article number, title, and the month and year of publication. If you also include a clear description of your problem or question we can often give you an answer straight away, without having to ask you for further details because you didn't make it clear just what you wanted to know in the first place. Elektor Labs says *thank you*!

(110131)

Internet Link

[1] www.elektor.com/contact

E-LABs INSIDE

A quick temperature measurement (2)

By Thijs Beckers (Elektor Netherlands Editorial)

What's impossible for an IR thermometer, such as those in last month's test, is a piece of cake for a so-called thermal camera, which is also known as an IR camera. Prompted by our test, Vincent van der Walle (**photo 1**), a sales engineer from FLIR distributor KWx B.V. [1],[2], came to us to explain thermal cameras. This, of course, included a demonstration of a number of the thermal cameras developed by FLIR. Although the cheapest model, with a price tag of just under €/£1000 (the FLIR i3), is perhaps a bit too expensive for the enthusiast but when used in research and development departments even the more expensive model will pay for itself in a very short time. How? Continue reading...

Foremost it is important to understand what you are measuring. This was already highlighted in the IR termometer test published in last month's issue of Elektor. In particular, the emissivity of an object has a significant influence on the measured tem-

perature (also refer to the sidebar 'Emissivity' in last month's test article). Something that is difficult to see with an IR-thermometer but is easily demonstrated with a thermal camera: measuring the temperature of a simple (soup) tin, filled with water that has just been boiled, will quickly show the 'problem' when using an IR-thermometer (see **photo 2**). When the emissivity is not adjusted to match the material (in this case a tin can with an emissivity of about 0.15), the measured value deviates so much that it is essentially useless. Our reference model, the Fluke 572, indicated a value of 34.5 °C when the emissivity was not adjusted.

With an IR thermometer this measurement error is difficult to spot. But with a thermal camera (which essentially uses much the same sensor as that in an IR-thermometer, but many of them and arranged in a matrix) it is immediately obvious that something 'strange' is happening (see **photo 3**). The tin appears to be reasonably cool (the measuring spot in the centre reads 28.9 °C), while the water and the pieces of electrical tape on the side are shown very bright; about 72 °C, when referring at the scale at the bottom of the display.

What the camera actually 'sees' when it is pointed at the tin is the reflected heat-emission from the surroundings of the tin and part of the heat of the tin itself (that's why the temperature is a little higher), instead of (only) the temperature of the tin by itself. However, the pieces of coloured electrical tape do give the correct result. The actual colour of the tape is not important, as can be seen in the photo. Only the material and the specific emissivity play a role (and this is about 0.95 for the pieces of tape). So, with an IR camera you can very quickly see where the measurement is going wrong and where it is going right.

But despite this, it is nevertheless a good idea to remain cautious and you still have to understand what you are doing. For example, we pointed the IR cameras under test at the heatsinks of a dual output power supply, one half of which supplied 2 ampères into 8 ohms and the other half 0.5 A (also into

1

2

3

8 ohms). The heatsinks had quite a matt surface and while the manufacturer had selected different colours for the heatsinks, the IR cameras had no problem showing the correct image of the heat distribution across the heatsinks (see **photo 4**). The heatsink on the right is clearly much warmer and almost appears to glow on the camera screen (this heatsink was about 55 degrees).

What we did notice however, is that the top of the heatsinks appeared to be much cooler (even cold) when we held the camera at the same level as the power supply (that is, looking straight at the back of the power supply). What's happening now? The top of the aluminium heatsinks acted as a mirror and reflected the temperature of the relatively cold wall behind the power supply (just as an asphalt road surface on a hot summer's day sometimes behaves like a mirror when you look into the distance).

The same effect is clearly visible in **photo 5**: The thermal image of the tin can with hot water is mirrored by the surface of the table, while the table has a higher temperature only directly underneath the tin can.

Such a measuring error can only be noted — and cannot be avoided. It is not possible, unlike when making sound measurements, to have a kind of 'dead room' where you can measure thermal radiation to prevent these types of reflections. Every object emits thermal radiation and influences the measurement. That is why you have to understand what you are actually measuring and with an IR camera the result is much easier to interpret than with an IR thermometer.

What kind of applications does the design department have for an IR camera, you may perhaps wonder. The answer is simple: IR cameras can very accurately determine the temperature of any object that appears in front of its lens. This is very practical when looking for problems on a printed circuit board. Point the camera at the board and you can see immediately where the temperature is (too) high and where there is, in all likelihood, a (design) fault. The FLIR T425 in photo 6 was so accurate, that we could see the 'IR-fingerprint' of the silicon inside the FV1-chip in the digital multi-effects unit (Elektor September 2010) through the top of the package, even though the temperature

was only a few degrees above the ambient temperature (chip temperature: 27 °C). And for the observing reader: towards the left of the image you can see that the two voltage regulators have also acquired quite a bit of colour.

When you've had enough of looking at electronic circuits, you can also imitate CSI with such a camera… A hand print on a table will leave a thermal trace for at least five minutes, which is easily seen with such a camera. Footsteps are often easily traced, particularly on carpet. Follow your colleague who just walked past and try to find out where he or she just came from…

You can often also see exactly which part of a dropped object hit the ground first and where on the ground it hit. This is really just simple physics, when the object hits the ground the kinetic (moving) energy is converted into heat. And this can be easily seen on the IR camera.

(110273-I)

Internet Links

[1] www.kwx.nl
[2] www.flir.com

5

4

6

One-eared Skype

By Thijs Beckers (Elektor Netherlands Editorial)

When we tested the Microphone Conferencing System described elsewhere in this edition we were initially satisfied with the results. The left and right microphone signals entered the PC as expected. Several tests using Windows Live Messenger went smoothly and we could be understood clearly by the people at the other end. Everything appeared to work as it should.

However, when we had our French Editor on the line via Skype, using the laptop belonging to the International Editor-in-Chief, things started going wrong. Only one microphone worked. Was there a cable fault? Was the sound driver on the laptop installed properly? Was it a setting in Skype? Perhaps a 9 V battery was flat? But no, all these were in order and worked properly. To be completely sure, the cables were changed with known working ones, but the problem persisted.

Then we swapped the modules over and we noticed that the other module stopped producing sound. This got our thoughts going. With Windows Live Messenger everything worked perfectly. With Skype only the left input channel worked, and swapping the modules didn't make any difference. A quick search on the Internet made it clear that we weren't the only ones who came across this 'feature' in Skype. It was just impos-

sible to send a stereo signal with Skype. Granted, Windows Live Messenger also sent a mono signal, but at least it did things properly and combined the left and right input channels. If the various forums are to be believed, Skype has failed to do this since at least 2005! We're sure they had a good reason for this... Anyway, in order to use our system with Skype we had to connect both channels to the left-channel input of the PC or laptop. The simplest solution is to use two resistors to combine the signals. This isn't the most elegant solution, but at least it's effective. A quick test confirmed that it all worked properly, and an inset in the article explains how to make such a 'Skype-cable'. All's well that ends well!

(110314)

Problems with noise

Thijs Beckers (Elektor Netherlands Editorial)

Generally speaking, audio-oriented enthusiasts tend to dislike noise and would rather see the back of it. However, for once it was the opposite when we designed the Wave Sound Generator published elsewhere in this edition. This design required a transistor that produced as much AF noise as possible. If you refer to the circuit diagram, you'll see that T2 is being 'misused' as a noise source.

The original circuit from 1996 made use of a BC547 and a 12 V power supply. Today, Chris Vossen of Elektor Labs preferred to use a lower supply voltage allowing the circuit to operate from a 9 V battery. One problem he ran into was that the noise generating transistor T2, which was replaced in the new circuit with an SMD version type BC847. This required, just as for the original BC547, a minimum of 10.5 V in order to generate a decent amount of noise. Consequently a simple 9 V battery could not supply a sufficient voltage. We also tried using a BC850, which requires a slightly lower voltage to work as a noise generator in this circuit, but unfortunately the 9 V supply was too low for this as well.

We could of course have made some drastic changes to the circuit, for example adding a boost converter to increase the 9 V supply voltage, or using a zener diode and designing a completely new circuit around that, but the editorial, the DTP and the printer were all telling us to hurry up because there were deadlines to meet, so in the end we had to cut the knot... no 9 V supply then. Oh well, at least with an AC adapter you don't have to worry about running out of power.

(110280)

Here Comes the Bus! (5)
We're off!
The first circuit boards and software
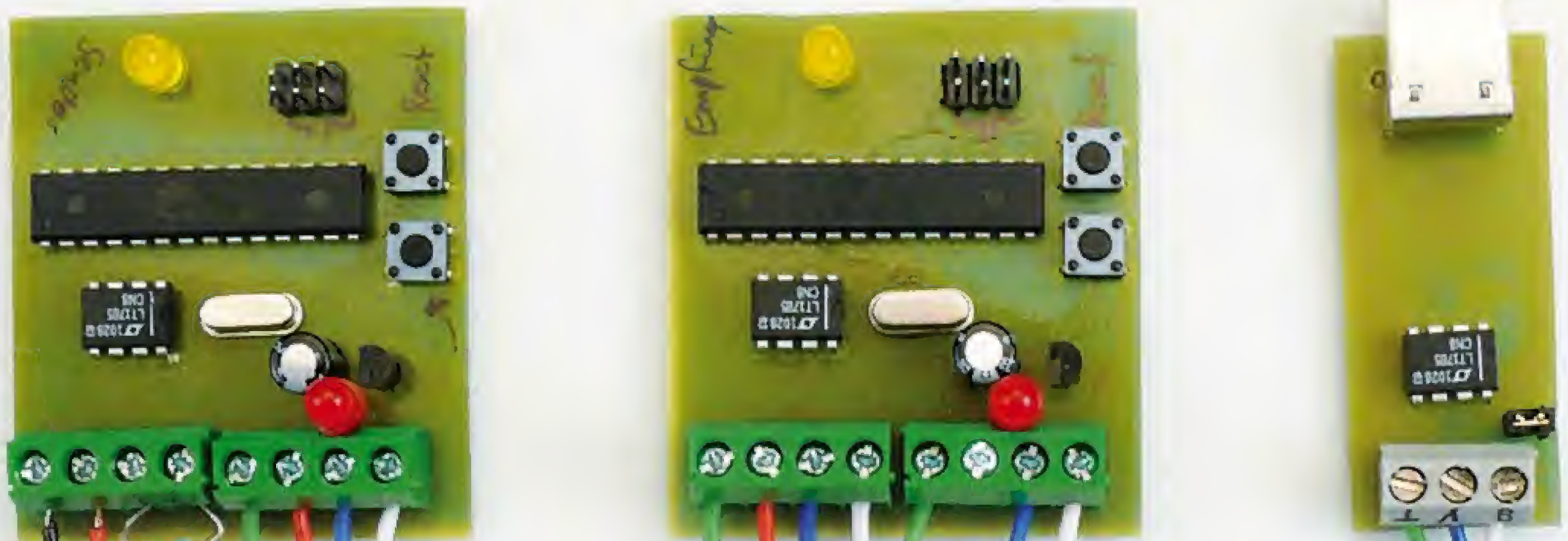
After all the theory we covered in the previous articles in this series, we are now finally ready to try to send some real bytes over some real wires! The test hardware takes the form of two ATmega88 test nodes and a compact USB-to-RS-485 converter, and to these we add a dash of BASCOM and Visual Basic. Oscilloscopes at the ready! As always, we find that there are lessons to be learnt as we lay the foundation stone for future hardware and software developments.

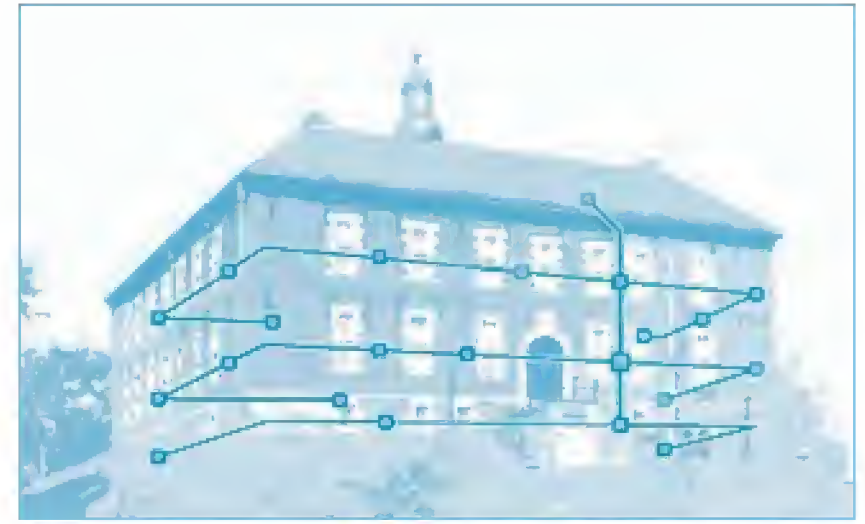By Jens Nickel (Elektor Germany Editorial)

In the previous instalment in this series we described a very simple bus communication protocol, where a message just consists of 16 bytes [1]. The first byte always has the value 10101010 binary = AA hex = 170 decimal and can be used for synchronisation. The second byte determines where the address and data payload are located within the packet and whether the last two bytes are used for error detection. That's that sorted out then, I thought as I sat back after polishing off the previous article; now I can relax for a bit. How deluded I was!
The newly-set-up mailing list for interested readers (including some highly experienced designers) was not going to let me rest. Readers were chipping in with ideas and suggestions drawn from their own or other people's bus designs. Since RS-485, as

normally defined, does not include a built-in mechanism for detecting bus collisions, there was a lively discussion on the topic of how we would avoid collisions in our system in the future. Scheduling was a frequently-aired idea, whereby a special network node would be responsible for allocating 'transmission slots' to the other bus participants. The simplest form of this seemed to me to be a kind of query-response structure, with the scheduler node sending a message to another node to prompt it to transmit, the node in question then sending a suitable reply. The scheduler's attention would then turn to the next node, and so on. This approach, which I dubbed 'round-robin ping-pong', gives rise to a problem, however: if 32 bus participants are to be interrogated in turn, then each one will only be

given a chance to speak about once per second (assuming our low-speed bit rate of 9600 baud). Another topic of discussion was robustness (what to do if the scheduler fails?) computing time in the bus nodes (each must be continuously listening) and the rather tricky problems that arise when you try routing messages between two interconnected bus segments.

The first prototypes
From my own experience I was aware that during a theoretical discussion like this it can be useful to do a few experiments, and that when you see something actually working for the first time, it is a great motivational boost. So we must press on with our first prototypes and software! We could come back to the question of schedul-

ing later, and in any case we had already discussed possible solutions to the problems mentioned above. For now, we just wanted to send a few bits over some physical wires. Otherwise I would have had to put things off for a while, as the editorial team was in the middle of putting the April issue together and our labs were about to be inundated with other projects.

Fortunately we had on our ElektorBus team a highly experienced engineer in the form of Günter Gerold, who could not wait to get to work on the project. He's the kind of chap who rarely lets his soldering iron get cold, and who thinks nothing of etching a couple of printed circuit boards. So, just a few days after we had sent him a couple of components, a parcel arrived on my desk containing two fully-populated boards corresponding to the circuit diagram we gave in the previous issue. Not only that, Günter had also designed, etched and populated a small USB-to-RS-485 converter board! And, as if even that wasn't enough, he had also written a couple of lines of BASCOM to test the boards by sending a byte over the bus. This was accompanied by some test software in Visual Basic to check that the PC could also receive the byte via the converter board.

Test nodes and USB converter

The circuit board layouts are included in the download available from the web pages accompanying this article [2]. Populating the boards is not too difficult, although the design does reflect Günter's fondness for SMD components. The circuit diagram for the test node is repeated in **Figure 1** (and the photograph at the start of this article shows the prototypes). We used a standard two-by-three header for in-system programming (ISP): note that it is of course important to fit the AVR ISP connector the right way around.

The circuit for the USB-to-RS-485 converter (see **Figure 2**) is based on the one given in the datasheet for the FTDI FT232R USB-to-TTL converter IC [3]. The circuit is not too difficult to understand, especially when compared side-by-side with the circuit of the test node. In this case the DI and RO pins of the RS-485 driver are connected to

the TXD and RXD pins of the FT232R, which deals with all the details of USB communication. The FT232R's CBUS2 pin by default brings out the TXDEN signal, which we can use to provide half-duplex operation on the RS-485 bus. When data are received
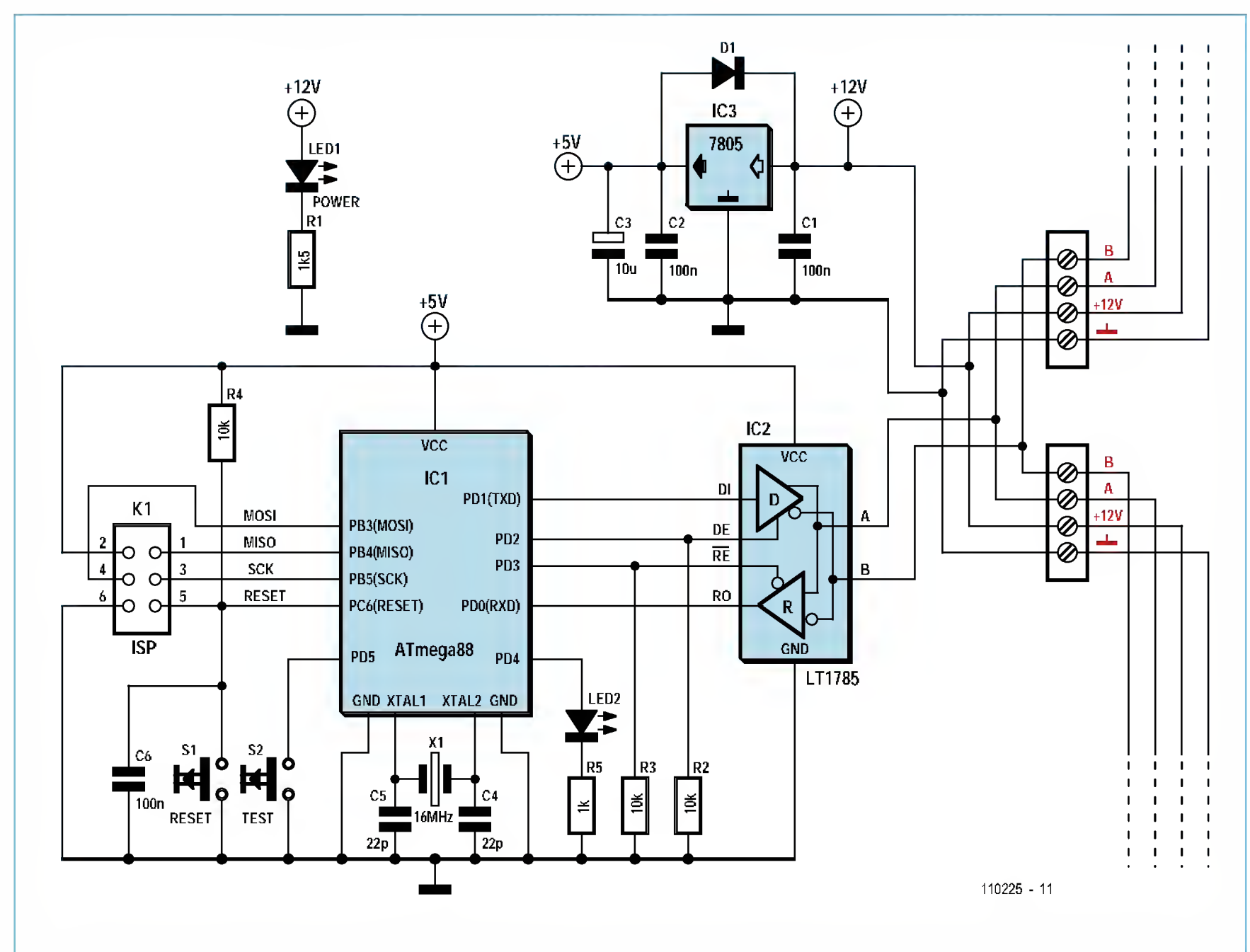

Figure 1. Circuit of the test node, as given in the previous instalment in this series. Two screw terminals are available for each bus wire, making it easier to extend the bus from node to node.
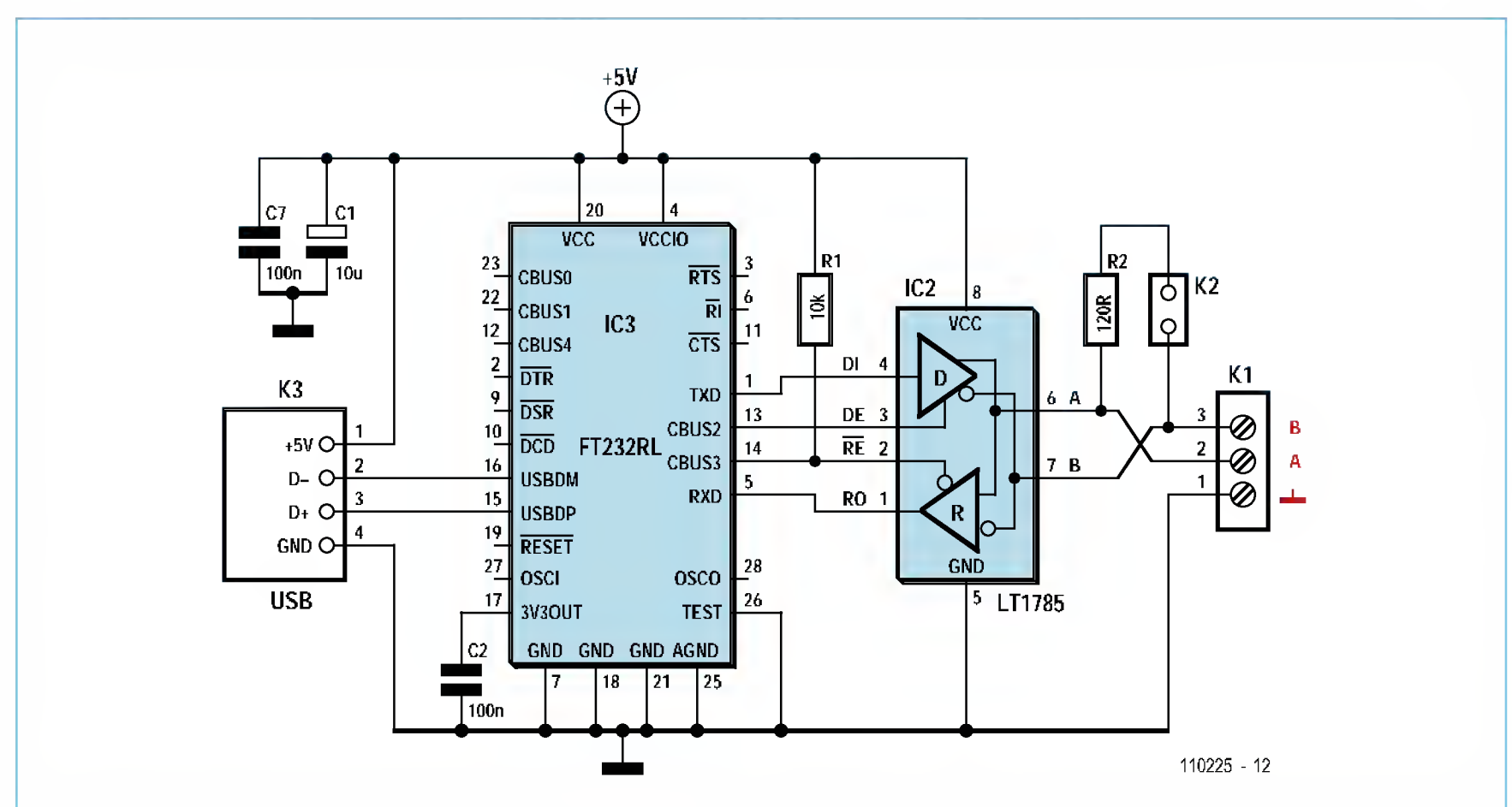

Figure 2. Circuit diagram of the USB-to-RS-485 converter based on the ever-useful FT232R.

Listing: BASCOM software (excerpt)

```
Readeeprom Ownaddress , 2        'Addresses in EEPROM
Readeeprom Otheraddress , 3


Do

 If Button = Pressed Then       'Button alias Portpin PD5
    If Buttonstatus = Released Then Sendeventflag = True
    Buttonstatus = Pressed
 Else
    Buttonstatus = Released
 End If

 If Sendeventflag = True Then
   Driver = Enabled          'Portpin PD2=1
   Receiverstop = Enabled    'Portpin PD3=1

   Set Ucsr0a.6              'flag (see below) must be (re)set

   Sendmessage(1) = 170           'Byte 0 in ElektorMessageProtocol (EMP)
   Sendmessage(4) = Otheraddress  'EMP Byte 3 = address-byte of receiver
   Sendmessage(6) = Ownaddress    'EMP Byte 5 = address-byte of sender
   Sendmessage(7) = 10            'EMP Byte 6 = first databyte of sender
   Printbin Sendmessage(1) ; 16   'send 16 bytes message

   Do
    Loop Until Ucsr0a.6 = 1   'wait until all bytes sent

    Driver = Disabled
    Receiverstop = Disabled

    Sendeventflag = False
 End If

 If Receivedeventflag = True Then      'an incoming message?
    Commandbyte = Receivedmessage(7)   'first data-byte

    If Commandbyte = 10 Then
       Toggle Testled
    End If

    Receivedeventflag = False
 End If

 Waitms 10

Loop


Onrxcomplete:         'Interrupt on first incoming byte

 Startbyte = Udr
 If Startbyte = 170 Then
    Inputbin Receivedmessage(2) ; 15   'Read following 15 bytes
    If Receivedmessage(4) = Ownaddress Then
       Receivedeventflag = True        'incoming message
    End If
 End If

Return
```

over the USB connection the DE pin of the LT1785 is taken high so that it drives the RS-485 bus. We used the same pin arrangement in the 'modded' full-duplex USB-to-RS-485 converter that we described in the December 2010 issue [4]. The CBUS3 pin gives us access to the PWREN# signal, which is low during normal USB operation [3]. Hence the receiver in the LT1785 is always active and the USB chip is always listening to the bus.

The jumper allows a 120 Ω termination resistor to be fitted between the A and B bus lines. The bus itself is connected via screw terminals, and the third contact of the terminal block is the ground connection for the board, and hence also for the USB connector, which is normally at earth potential. The significance of this connection (under normal conditions the bus will work without the earth) is described in more detail in the text box.

Fitting the FT232R requires a little delicacy and soldering know-how: it is only available as an SMD component (**Figure 3**). Elektor may produce a partially-populated version of the converter board at some point in the future; alternatively, it is possible to make use of the popular USB-to-TTL adaptor cable available from FTDI, which already includes an FT232R device [5].

First tests

For our first tests we wired the nodes together using ordinary wire (see lead photograph: bus line A is blue, B is white, +12 V is red and earth is green). The doubling up of the screw terminal blocks makes things simpler here, as one of the nodes can easily be wired directly to the USB-to-RS-485 converter. If we consider the PC to be bus 'node 0', then we can call the two ATmega test nodes 'node 1' and 'node 2'. We need to make sure that there are termination resistors at either end of the bus: at the converter end, this is just a matter of fitting the jumper; at the other end, we can simply fit a (leaded!) resistor between the spare A and B screw terminals on the second test node. For experimentation we need an AVR ISP-compatible programmer and a development environment. Since Günter had already put together a couple of lines of

BASCOM, I decided to stick with this environment for now, though I promise to add some C code later on. Installing BASCOM is not difficult, and the manufacturer's site includes a good set of instructions [6]. Incidentally, the free version of BASCOM, which is limited to a code size of 4 Kbyte, is more than adequate for our purposes: for comparison, the microcontroller has 8 Kbyte of flash memory of which the firmware described below occupies only about 10 %. I used an original Atmel AVR ISP mkII programmer. To get BASCOM to work with this device, it is necessary to install the libusb driver (which is open source and also free) [7]. The BASCOM documentation [6] also includes a guide showing how to do this.

Next I turned my mind to writing a first version of the firmware capable of sending and receiving 16 byte messages according to the *Elektor* Message Protocol (EMP). Then I wrote some software for the PC, using Visual Basic (the Visual Studio Express development environment from Microsoft is also free [8], but if you prefer, you could use the open source Sharp Develop IDE [9] as Günter does). Some help in installing Visual Studio can be found at [10].

While working on this code I encountered a couple of anomalies: sometimes the PC would suddenly receive a stream of phantom data bytes; at other times the bus would work only when the programmer was disconnected; and still other times it would only work when the programmer was connected! After some head-scratching, my colleague Thijs Beckers (who already has a couple of audio projects under his belt) and I finally ran this puzzling problem literally to ground: see the text box 'Grounding for Dummies'.

Firmware

The results of our efforts are available for download from the website as source code. The same firmware runs in the two nodes: it allows a node to send a 16 byte message over the bus when its test button is pressed. In accordance with the EMP, the message consists of the following bytes (see **Figure 4**): a start byte (170 decimal); a mode byte (zero); two bytes each for the receiver and transmitter addresses; and

finally the payload data. The CRC is not yet implemented, and so for simplicity's sake we have set the two bytes to zero. The payload consists of a byte with the value 10. Each node's own address, as well as the addresses of the other nodes, must be programmed permanently into the EEPROM of the ATmega using BASCOM: set bit G in the high fuse byte to zero ('preserve EEPROM') so that the EEPROM is not overwritten by subsequent firmware updates.

The listing shows an excerpt from the first version of the firmware. When a node receives the first byte of a message, it triggers a *URXC* (*UART RX complete*) interrupt in the microcontroller. The program then jumps to the label *Onrxcomplete*. The subsequent lines of code read the next fifteen bytes from the interface using the *Inputbin* command and store them in the byte array *Receivedmessage*. Then a flag bit internal to the program (not to be confused with the ATmega's flag register) called *Receivedeventflag* is set. This acts as a signal to the main program that a message has arrived and that data are waiting to be processed.

The main program itself consists of an infinite loop with three sections of program code within it. The first section checks whether the button has been pressed. If so, the dedicated *Sendeventflag* is set, which means that, as a result of an internal event, there is a message to be sent out over the bus. A further flag bit, *Buttonstatus*, is used to save the state of the button to prevent sending multiple messages if it is held down.

The two remaining sections of program code check whether *Receivedeventflag* or *Sendeventflag* are set. In the former case the received message is processed. If byte 6 of the message is equal to 10, the test LED on the node is toggled. (Note that BASCOM indexes arrays starting from 1, so byte 6 has index 7 in the byte array.) If *Sendeventflag* is set a message is assembled for sending. The correct address bytes for the receiver (i.e., the address of the other node) and the transmitter are retrieved from the EEPROM as mentioned above. Before the sending can commence the DE input of the bus driver must be taken high: in BASCOM this is done using the command *PORTD.2 = 1* (for more information, see the short BASCOM course
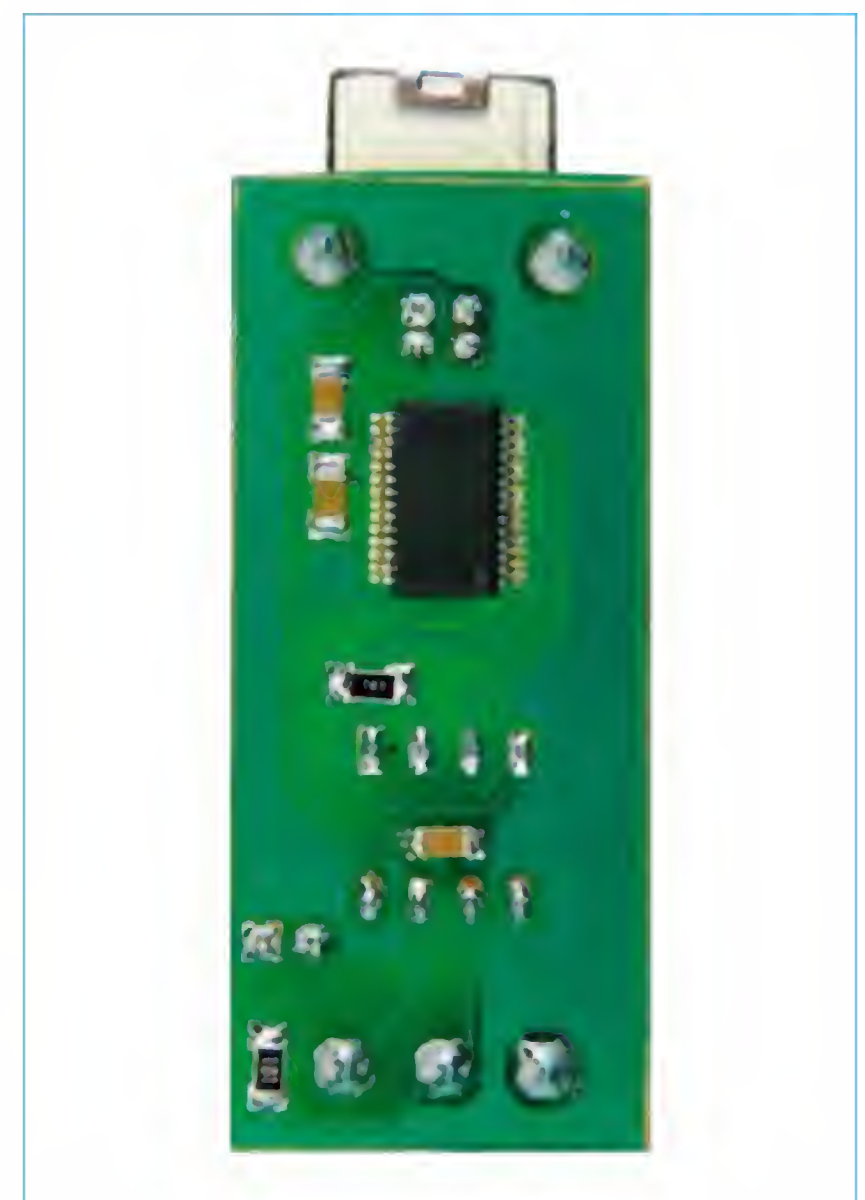

Figure 3. Soldering the FT232R is no easy task, and we are looking into ways to make construction easier for beginners.

at [11]). To avoid having to refer constantly to the circuit diagram, I have set up some constants and aliases at the start of the program. This makes the command rather easier to read: *Driver = Enabled*. The message is sent using the *Printbin* command, and then a small loop waits for all the bytes to be drained from the transmit buffer (bit 6 of the *UCSR0A* register in the Atmega88 is
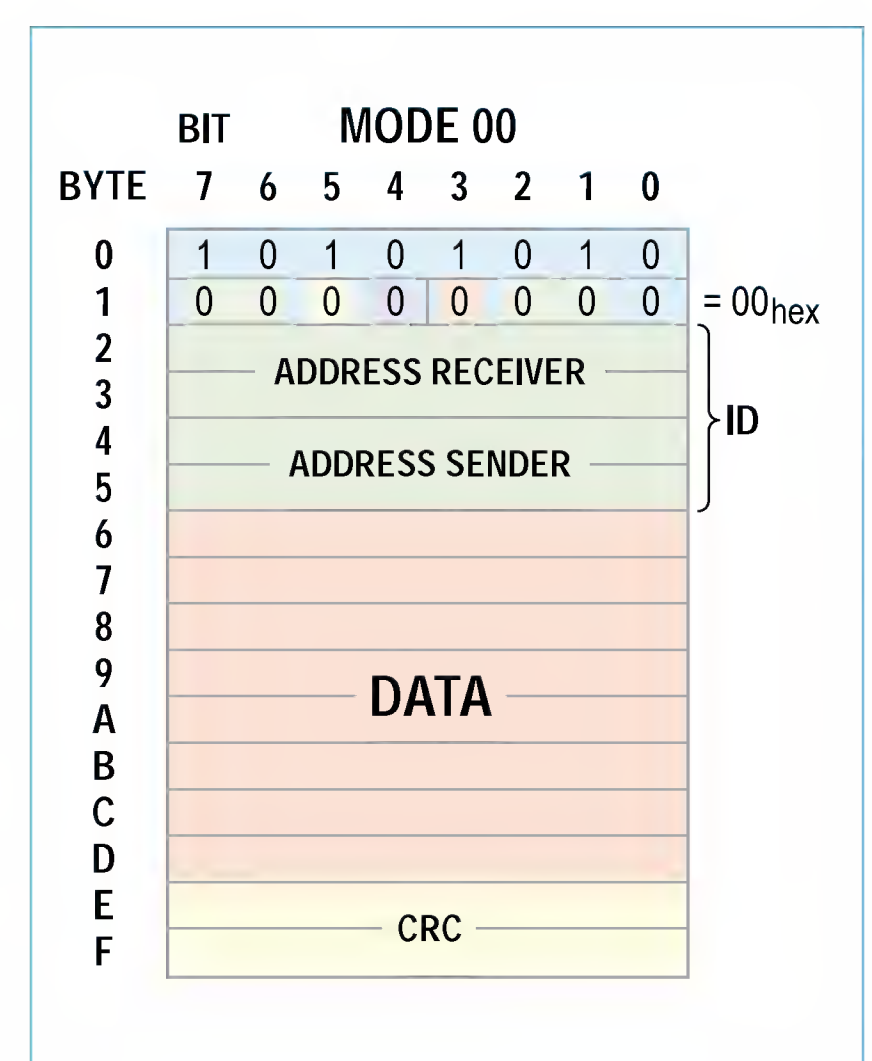

Figure 4. The structure of a 16-byte message using mode 00 (hex) of the Elektor Message Protocol. The CRC field is not yet implemented.

Grounding for Dummies

It took me a good hour or two to realise that it was essential to wire the ground connection on the bus (green wire) to earth, and, just as importantly, not to earth either of the two signal wires (blue and white)!

The latter can easily happen if you unthinkingly attach an oscilloscope across the bus to have a quick look at what is going on. What is the problem? The LT1785 bus driver can only drive the two bus lines between 0 V and +5 V relative to its ground, and represents one bus state by a positive differential voltage (taking one differential output High and the other Low) and the other bus state by a negative differential voltage (taking the first differential output Low and the other High) [16]. When the bus is idle the two wires typically both float to a 'common mode' voltage of about 2.5 V (relative to the driver's ground), with approximately zero differential voltage between them. Now, attach an oscilloscope probe across the bus wires and a short can be created via the ground clip on the probe, AC power earth, the PC's earth connection and hence its USB ground connection, and the ground point on the USB-to-RS-485 converter board. The driver will no longer be able to take the output corresponding to the shorted bus wire to +5 V relative to its ground, and it will therefore not be possible for it to establish one of its two output states.

The ground screw terminal must be connected to the bus ground at each node, and hence to the ground of all the other bus participants. An interesting experiment that we inadvertently tried is to power the test nodes independently from common-or-garden (switching) mains adaptors whose outputs are not connected to AC grid earth. The ground output of these devices floats (or, more accurately, oscillates freely at 50 Hz) relative to AC grid earth, and hence so does the common mode voltage of the bus. As long as the only connection between the nodes remains the bus, everything will still work; but woe, when the properly-earthed USB-to-RS-485 converter is plugged in! The oscillating voltages on the bus are interpreted by the receiver as bits and bytes, and all the PC software sees is junk. Connect the earth between the converter and the other nodes and all will be well again.

The idea of using a third screw terminal came from John Dammeyer. He points out that the earth connection also has a protection function: without it, if a wiring fault should mean that a high potential relative to mains earth appears on the ground connection (and hence on the bus wires) at one node, the only protection for the computer's USB connector is the RS-485 driver on the converter board. With the extra ground connection at mains earth potential, it is likely that at worst a track on one of the bus nodes will be damaged. Repairing that is a rather cheaper prospect than replacing the motherboard in the PC!

TXC). The DE pin of the LT1785 can then be taken low again.

PC software

To get communications working on the PC, the VCP (virtual COM port) driver for the FT232R chip has to be installed. This can be downloaded from the FTDI website [12], and there is an installation guide at [13]. With that done, it should be possible use a terminal program running on the PC to view the bytes received on the bus by the converter (although I have not actually tried this). Our first PC software is based on Günter's VB.NET program that he used to test the converter board he assembled. He had already worked out how to use the .NET *SerialPort* class [14] to configure a (virtual) COM port and receive characters. I also found the book 'Serial Port Complete' by Jam Axelson [15] very useful. In the book Jan, whom incidentally I was able to persuade to join our mailing list, covers the basics of COM ports, RS-232 and RS-485, with a wealth of practical tips and advice on isolation, cabling, termination and the like. Program examples on the PC side are equally practical, using the .NET framework throughout. Listings are given in VB and in C#.

A detailed description of the PC software is outside the scope of this article, but the source code is commented. *ComPort* is the instance of the *SerialPort* class that we use. Its *DataReceived* event [14] is used to trigger a call to the subroutine *Receiver*, which processes the incoming messages. We have used the property *ComPort.ReceivedBytesThreshold* to ensure that the event is triggered only when a total of 16 bytes, or one complete message, has been received: being able to do this is a handy consequence of our choosing a fixed message length. Unfortunately it is not possible to visualise the incoming message by chang-

---

Weblinks

[1]   www.elektor.com/110012

[2]   www.elektor.com/110225

[3]   www.ftdichip.com/Support/Documents/DataSheets/ICs/DS_FT232R.pdf

[4]   www.elektor.com/100369

[5]   www.elektor.com/080213

[6]   http://avrhelp.mcselec.com/index.html

[7]   http://sourceforge.net/projects/libusb-win32/

[8]   www.microsoft.com/express/Downloads/#2010-Visual-Basic

[9]   www.sharpdevelop.net/opensource/sd/

[10] www.elektor.com/100539

[11] www.elektor.com/080330

[12] www.ftdichip.com/Drivers/VCP.htm

[13] www.ftdichip.com/Support/Documents/InstallGuides.htm

[14] msdn.microsoft.com/en-us/library/system.io.ports.serialport.aspx

[15] www.lvr.com/spc.htm

[16] http://cds.linear.com/docs/Datasheet/178591fc.pdf

Figure 5. Screenshot of the PC software in action. Our mini-controller can switch the LEDs on the nodes and also show all the messages being sent on the bus.

ing elements in a form directly from the routine *Receiver*; instead, it has to be done in its own routine, called from *Receiver* using the *BeginInvoke* method. The received bytes have to be transferred to this new routine in an array. Here is the grisly explanation for more advanced programmers (we recommend less advanced programmers just put up with the situation and leave the relevant code alone!): the *DataReceived* event is triggered in its own thread, and changing user interface elements is only possible in the main thread. We therefore need to communicate the data between the two threads. The routine *ShowMessage* is responsible for processing and displaying the message.

Operation

After starting up the PC software the correct COM port must be selected in the combo box. Then press the *Connect* button to set up the connection. The software will now listen to the bus and display all received messages, along with timestamps, in the text box. Messages sent from the PC will also appear here. The two test nodes are depicted in the middle of the form, each with its test LED and test button. Pressing the 'node 2' button is equivalent to pressing the button on that node's board, and the LED on the other node should turn on or off.

The software can easily be extended. For example, I modified it to incorporate an acknowledgement from the receiver: when its LED is turned on, it transmits a message containing the byte value '11', and when its LED is turned off it sends '12'. This allows the actual state of the LEDs to be displayed on the PC, and marks the first step on the way to making a complete home automation controller. I'm sure you will quickly think of lots of further ways to extend and improve the system!

(110225)

What do you think?

Feel free to write to us with your opinions and ideas for the development of this system.
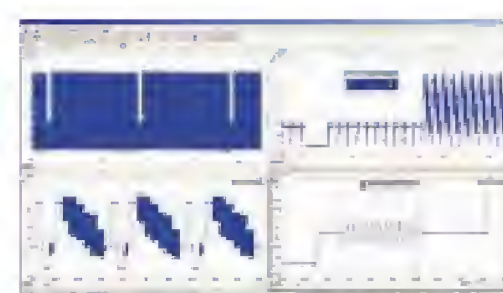
1-Channel DMX512 Light Dimmer

Developed using E-Blocks and Flowcode 4 for PICmicro

By Per Stegelmann (Denmark)

DMX512 is the industry standard for lighting equipment on stage and in theatres. Fortunately the standard is well documented so armed with this tutorial-style article nothing should keep you from building a DMX512 compatible dimmer yourself, perfecting it, find applications well beyond lighting and learn PIC and C programming along the way.

The DMX512 dimmer discussed here was developed out of the **DMX512 Tester** from Elektor June 2002 [1]. The tester proved a valuable device as you can select any channel you want to work with (up to 480), as well as change the 'Start of Frame' (SOF) character used, allowing you to test other DMX512 equipment besides dimmers.

The microcontroller used in the DMX512 Dimmer is a PIC16F88 with its built-in serial UART that's ideal for the DMX receiver function. The 16F88 also has the usual TMR0 timer/counter module, which is used here to control the triggering of the triac, which in the end governs the effective voltage applied to the load connected — observing the AC voltage zero-crossing of course with the help of a detector. When a zero-crossing is detected PIC timer TMR0 is loaded with a value that controls the triac's firing instant, where effectively 0x00 = zero power and 0xFF = maximum power. Actually, when 0x00h is written to the lamp, it is not fully quenched. Instead a small permanent glow is established to extend the lifetime of the lamp and make it respond faster. The ramp eventually governing the lamp brightness can be altered in the firmware.

Circuits and sub-circuits

The circuit of which the schematic is shown in **Figure 1** can be thought of as consisting of a number of sub-circuits identified as power supply, microcontroller, zero-crossing detector, AC power-line interface, DMX address selector and RS485 driver. Let's have a look at them.

Features

- BCD thumbwheel setup of DMX address – all 512 channels are selectable.
- LED indication of wrong DMX address selected. (slow flashing LED)
- Power-down of module on overheating
- LED indication of thermal power-down (fast flashing LED)
- All Flowcode v. 4 programmed
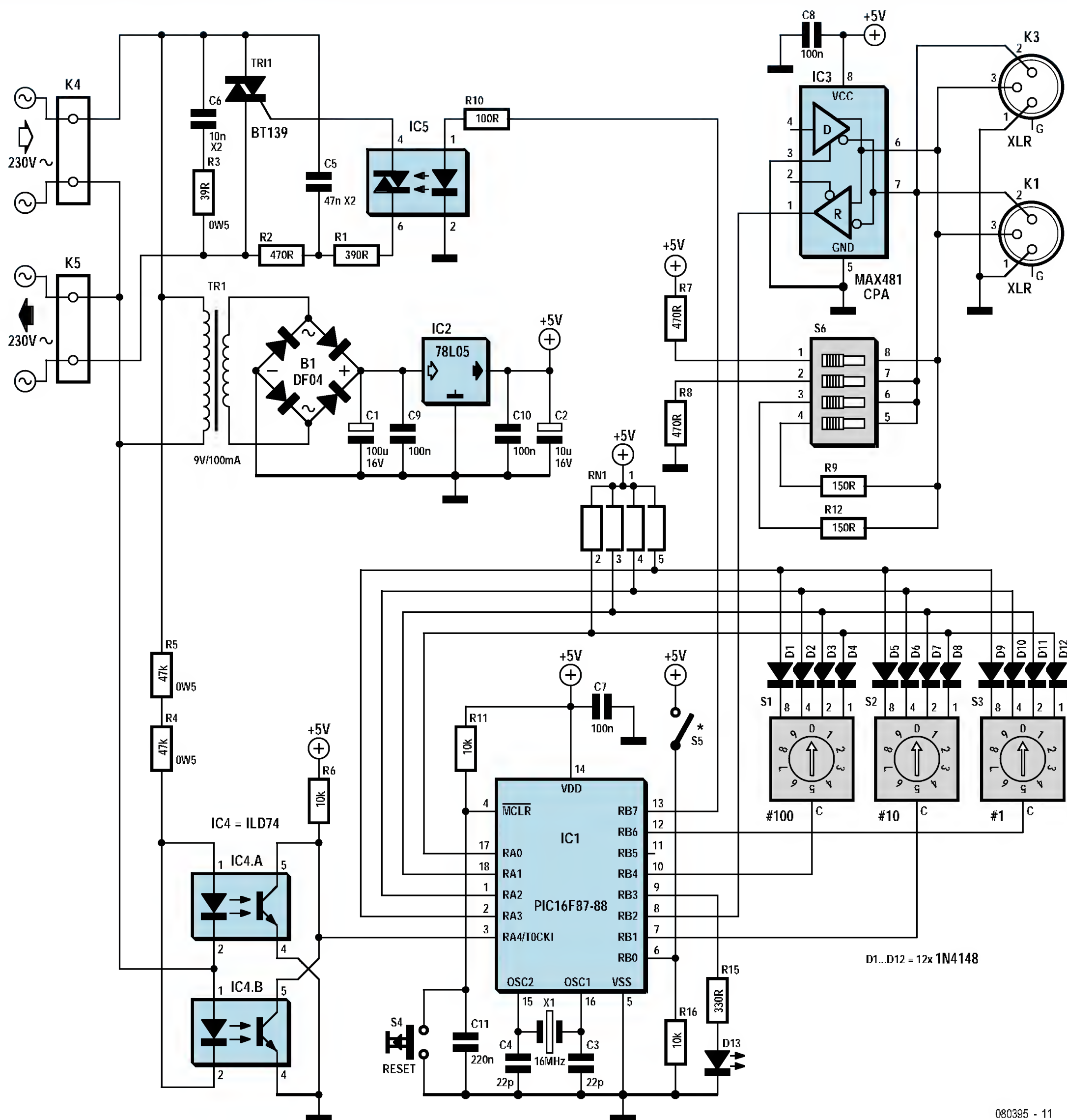
Figure 1. Circuit diagram of the Flowcode-developed DMX512 Dimmer.
Elektor USA readers please adapt relevant component values to match 115 V AC line voltage.

Elektor Products & Services

- E-Blocks PIC Multiprogrammer: # EB006
- Flowcode 4 for PIC4 Student version: # TEFLCST4
- Flowcode program (.fcf), C program, source code, hex files (free download): # 080395-11.zip
- Hyperlinks in article
- Items accessible through www.elektor.com/080395

Programming the microcontroller

The programming of the PIC 16F88 microcontroller is straightforward. If you have a programmer that works with Flowcode you just set up the chip as shown in the screenshot below and then press the **'Compile to chip'** icon. If you don't have a programmer that works with Flowcode you use the **Compile to HEX** option instead and then use your own programmer to program the chip.

A problem was encountered when a PIC was programmed for the first time. It looked like everything was fine, but it turned out the code was not programmed to the chip. After many desperate attempts the problem was identified as one setting in the PPP v3 options screen.

The culprit turned out to be the field **"Confirm HEX file format"**. By checking the box and saying No to PIC18 hex file when programming, it worked. Next, the tick was removed again, and the setup continued to work! What went wrong shall remain a mystery, but try this if you are having trouble when programming the chip, maybe it's only a W7 problem?

The power supply in straightforward with its power transformer (Tr1), bridge rectifier (B1), 100 µF reservoir capacitor (C1), 78L05 voltage regulator (IC2), smoothing and decoupling capacitors (100 nF and 10 µF; C9, C10, C2), Note Tr1 must be chosen to match the local line voltage (230 VAC or 110-127 VAC).

The microcontroller for this project is a PIC16F87/88 ticking at 16 MHz. It has an RC circuit connected to its /MCLR terminal allowing a reset button to be connected. Port line RB0 is used for the thermal protection function (S5 being the thermal switch), and TMR0 for zero-crossing detection and timing of the triac firing instant. Port line RB2 is the DMX data input pin, and RB5 the TX pin (not used here).

The zero crossing circuit consist of a dual optocoupler type ILD74 (IC4) and voltage droppers R4 and R5. The circuit detects both zero crossings.

The AC power circuit is built around a type MOC3023 optotriac (IC5) driving a BT139 power triac for control of the power to the load. This circuit can be traced back to a Fairchild application note.

The DMX address selector comprises three BCD switches, four pull-up resistors and 12 diodes. By multiplexing the common pins of the BCD switches, the value from each switch is read.

The RS485 driver is only one IC type MAX481CPA, a 4-way DIP switch and a couple of resistors. It's set up to receive data only.

Software development, step by step

While developing the firmware for the PIC in the DMX512 Dimmer the author went through several steps, these were documented and rendered below for you to follow and appreciate the ease of using Flowcode.

Step1. The code for reading the DMX channel address was placed in a loop that tested the switches and flashed the LED if the DMX address was not valid. An address must be in the range 1 to 512 to be valid. Once this was working, the function was turned into a Flowcode macro so it could be called as needed from the main program. The value of each switch is read by pulling a port B pin line Low, effectively enabling the connected BCD switch so that the value can be read via port A.

Step2. The next thing was the DMX receiver part, this was a bit more tricky to get right as it makes use of the standard Flowcode RS232 component and a C code box. The RS232 macro call is used to setup the baud rate for the receiver (250 Kbits/s) and to enable the hardware. The C code box (**Figure 2**) contains the code that takes care of receiving incoming DMX data and testing for errors. The DMX data format is defined as (in sequence and extremely simplified):

1. a Break on the DMX line (1 ms).
2. a Start of Frame (0x00 for dimmers).
3. the actual DMX channel values; this can be up to 512 data bytes but not all equipment sends all channels.

A Break condition on the DMX line is detected by testing for frame errors on incoming data (RCSTA = 0). When no more frame errors are received, the next data byte must be a Start of Frame (0x00), otherwise the receiver is reset (DMX_STATE = 0), and another Break on the DMX line is awaited. If there was a Start of Frame, the DMX channel count is set to 1 and DMX_STATE to 1. Now, when the selected DMX channel address is reached the DMX_VALUE is

updated with RX_DATA, the DMX_STATE is set to zero, and the receiver loop can start over again.

Step 3. With the receiver part working it was possible to proceed with the actual dimmer part of the software. This uses the TMR0 interrupt on overflow, and it's used for zero crossing detection as well as a timer to define the triac firing instant. This is done by means of a Flowcode interrupt component to enable the TMR0 interrupt, choosing TMR0 as interrupt source, T0CKI as clock source, the Low to High transition as the clock, and setting the prescaler to 1:1; see the screendump in **Figure 3**. Using a C code component to preset the TMR0 to 0xFF will enable the zero crossing detector as an overflow will occur at the first zero crossing of the AC line voltage. This in turn will call the macro called 'Fire_triac' containing a decision component at the start that decides which way to branch in the program. If Enable_trigger = 1 the triac will be fired, Enable_trigger is reset to zero, and TMR0 is ready to overflow on the next zero crossing. If Enable_trigger = 0 TMR0 is loaded with a value so that it will produce an overflow when the triac is due to fire.

Step 4. Soon, TMR0 value calculations were required. With the microcontroller running at 16 MHz, the clock cycle lasts 16 MHz / 4 = 4 MHz ~ 250 ns. Now the TMR0 prescaler is set to 1:256, giving 256 x 250 ns = 64 µs, being the basic clock for TMR0. If we now take 256 x 64 µs = 16.38 ms, that's the maximum period TMR0 can measure, which is more than enough here as you only need to count to approximately 10 ms (one half cycle at 50 Hz AC line frequency, if you use 60 Hz you will need to alter the software). Consequently TMR0 has to be preloaded with a value that will shorten the time, this value can be found as: 10 ms / 64 µs = 156, i.e. a preload value of (256 − 156) = 100 is called for. By adding this preload value to our DMX value (DMX value is divided by 2) we get the actual TMR0 preload value, this is calculated in a 'C code' box in Flowcode.

Step 5. The last thing that got added was the thermal protection circuit. This will protect against thermal damage using a temperature switch that closes at a temperature of about 60–80 degrees C, the switch proper being a normally open type. If not, the software must be changed accordingly.

Software structure

The software for the DMX dimmer was developed with **Flowcode 4 for PICMicro** and works as described below. The complete Flowcode program developed for the project is available free of charge from the web page set up for the project [2]. Got it handy? Here we go.

The program has three submacros, the first reads the BCD switches to obtain the DMX channel address to be used and the second is a TMR0 interrupt macro that detects zero-crossing and controls the triac firing. The last macro is a thermal shutdown feature, if called this macro will turn off power to the load; it is only called and activated if a thermal switch (S5, NO) is connected between RB0 and +5 V. Reading of the BCD switches is done at the start of the program, so if the channel setting is changed you will have to reset the
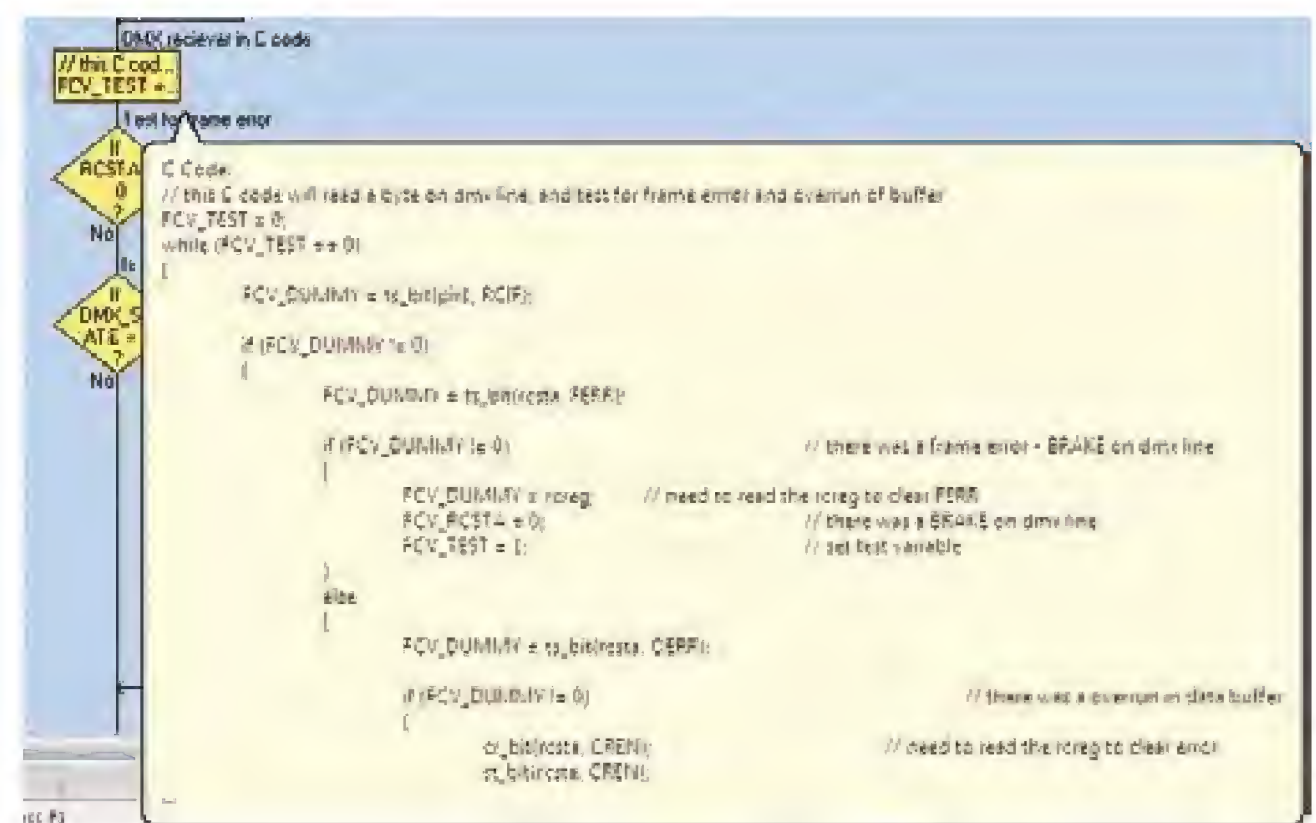
Figure 2. The C code box handles the DMX512 data reading and frame error / overrun detection.
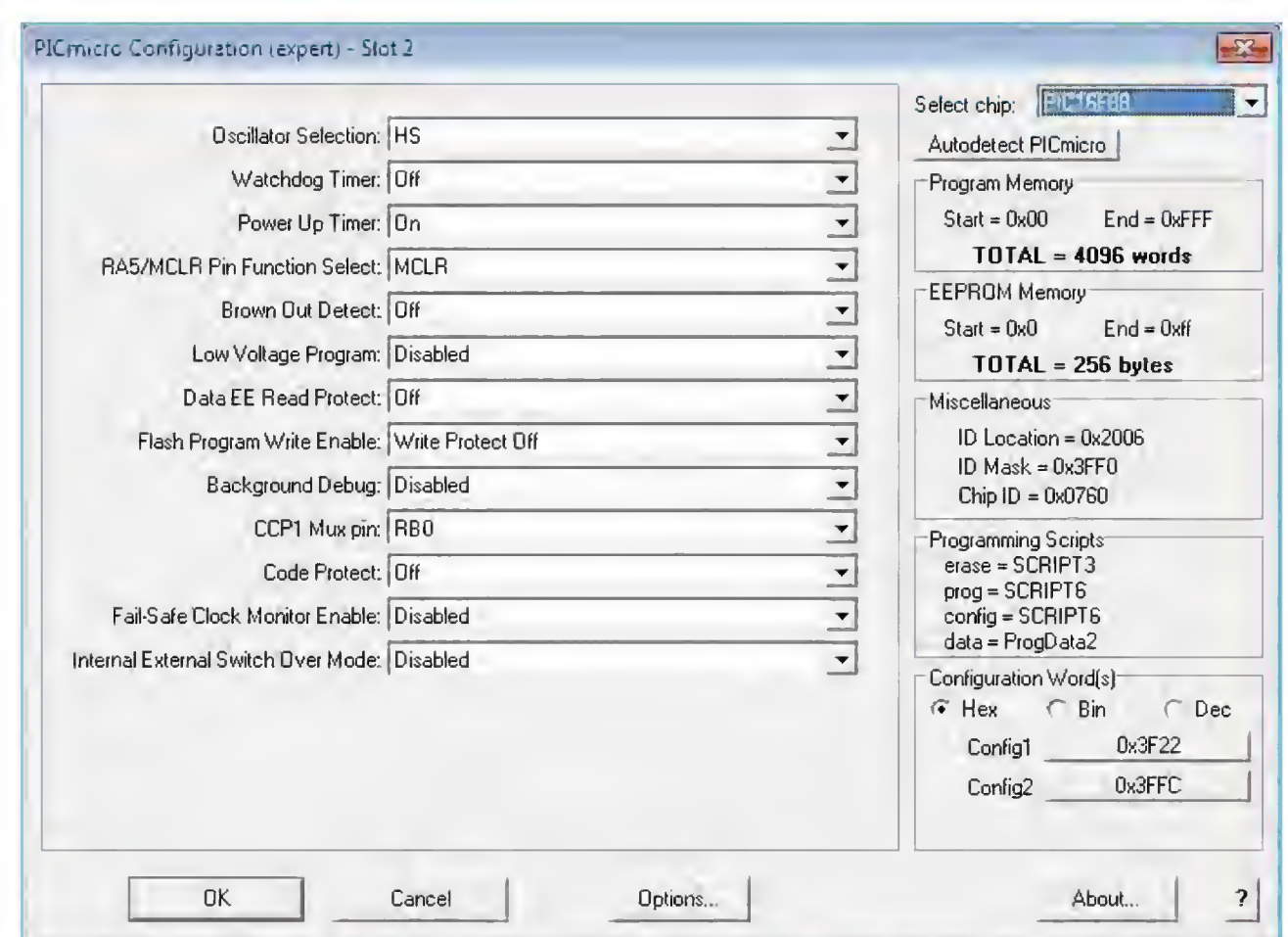

Figure 3. Flowcode method of configuring TMR0 for interrupts.

dimmer or switch the AC power to the module off and on again.

In the main program, the software is setting up the serial port by doing a dummy read. Now the DMX channel address is read and TMR0 is initialised to interrupt-on-overflow, as well as count on the rising edge input to pin A4. Also, TMR0 register is preset with 0xFFh, causing it to generate an interrupt when there is a rising edge on this pin — that's the way a zero crossing is detected.

Now the program goes into 'loop forever', this is where the actual DMX data byte receiving is performed (one of Apple's head offices in Cupertino CA, is at the address: *1, Infinite Loop*). First, the program has to detect a 'break' on the DMX line. This is done in the outer loop where frame errors on received bytes are detected. Once bytes without frame errors are received the next thing to test for is a Start of Frame (SOF) byte, which for dimmer modules equates to 0x00h. If it was a SOF byte, then the next thing to do is count the received bytes until the selected DMX address is reached, then copy the received DMX data to the register used in the TMR0 macro, that's it! A part of this routine is shown in **Figure 3**.
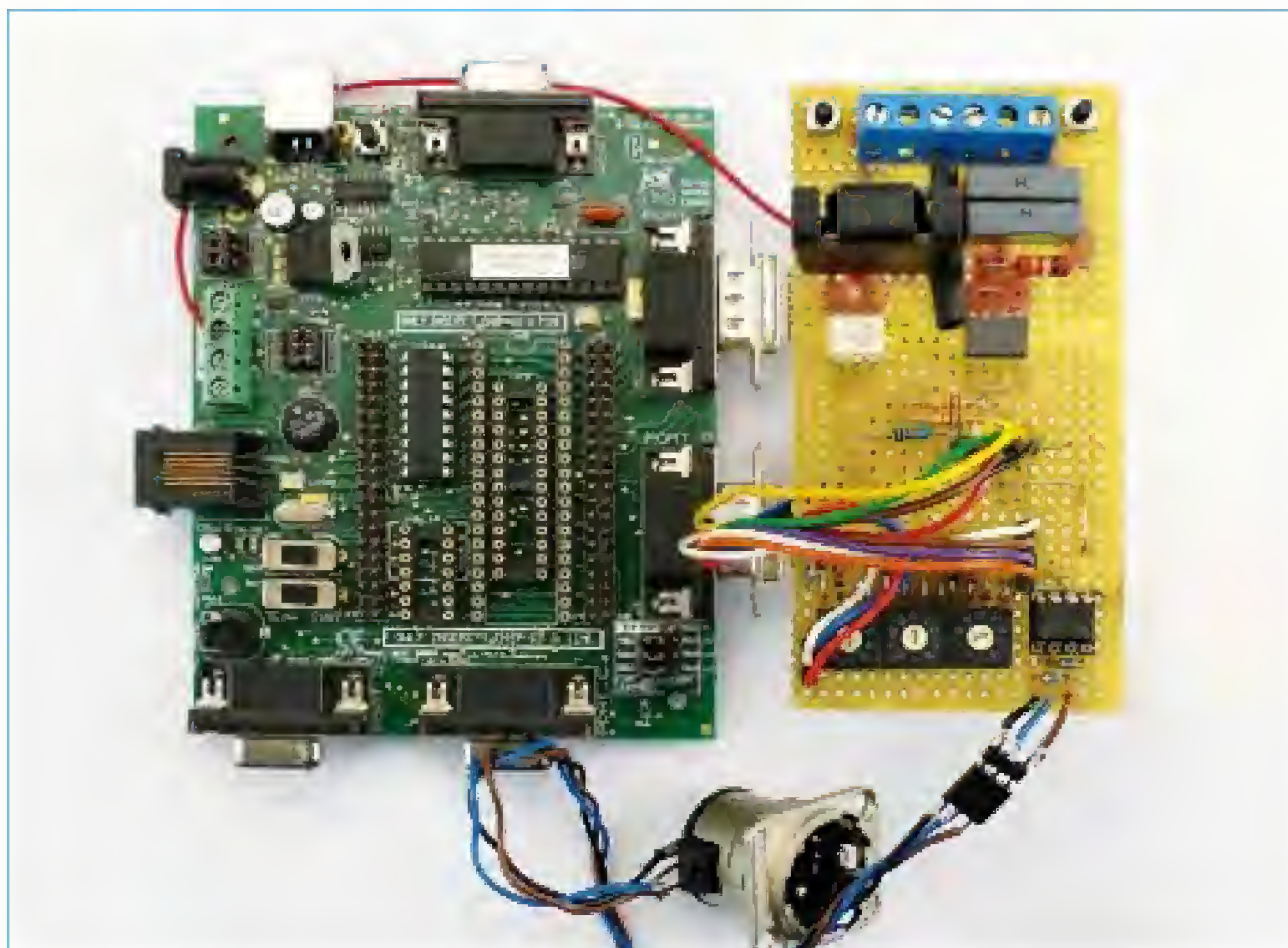
Figure 4. Experimental construction of the DMX Dimmer on a piece of breadboard connected to the E-Blocks PIC MultiProgrammer.
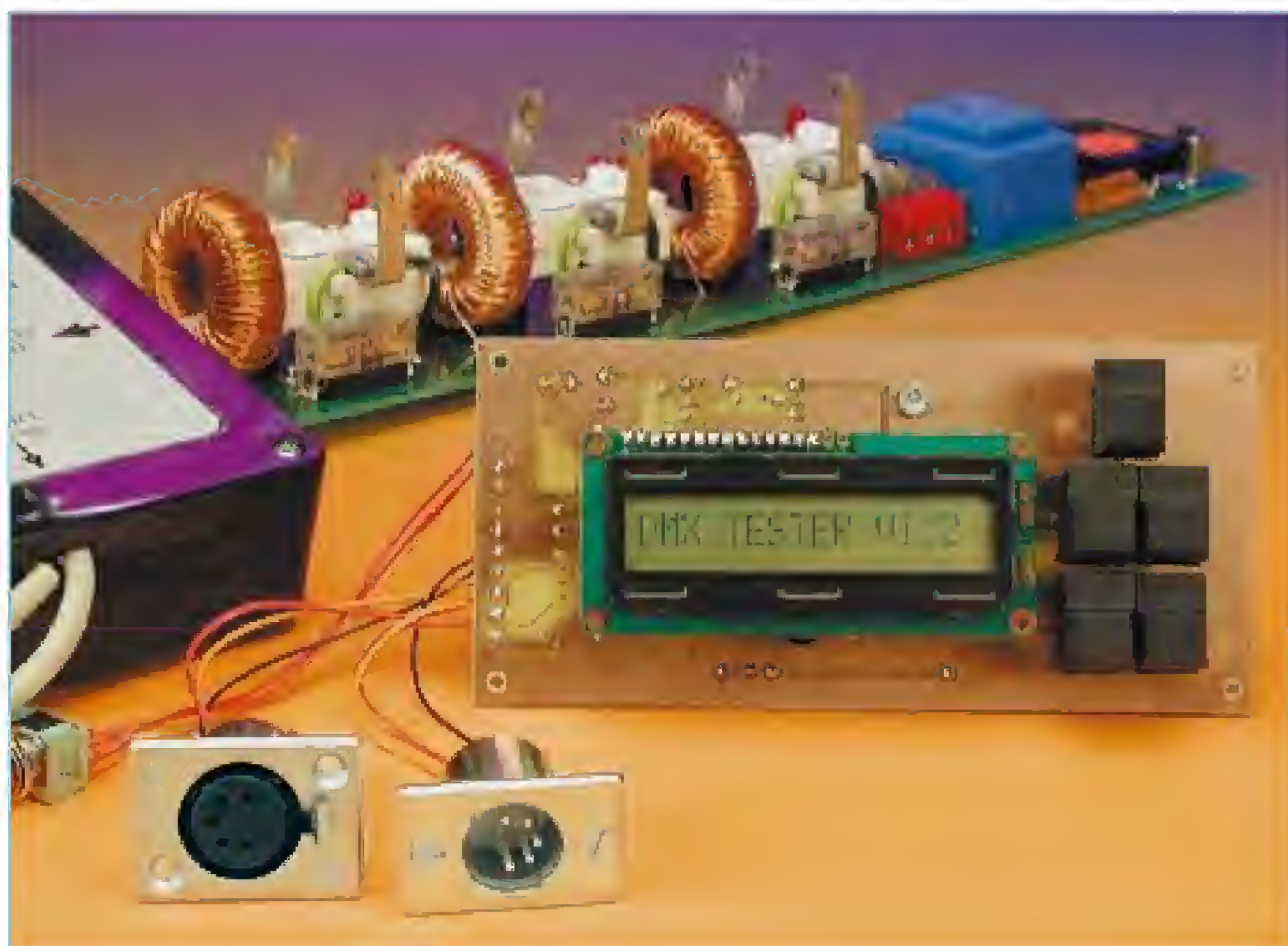

Figure 5. The Elektor DMX Tester described in the June 2002 edition.

trigger pulses to ensure proper trigging at any time in the AC half period. Now TMR0 is enabled again, this time for counting on the rising edge of A4 and preset with 0xFFh — this completes the macro.

Construction

Figure 4 shows the first setup for testing the idea. The perfboard (breadboard) contains a zero-crossing detector, AC line interface, RS485 interface, and three BCD switches for setting the DMX channel address. A LED is connected on port B (PB.3) on the sub-D connector, it's used for debugging and status display. The board and the EB006 MultiProgrammer [3] are powered externally by a wallwart with 9 V 250 mA output. The breadboard circuit was used for various experiments, not only with PIC microcontrollers but also Atmel and other devices, as it is fully isolated from the AC line voltage via optocoupler and a optotriac driver. For good measure we should add that no PCB was designed for this project by Elektor Labs.

The experimental dimmer board is connected to the E-Blocks PIC MultiProgrammer using 9-pin sub-D plugs connected to port A and port B. The Elektor DMX tester is pictured in **Figure 5** mainly to recall another brilliant Elektor design.

What enclosure to choose for this project will depend mostly on the conditions under which the dimmer is to be used. If it is for indoor use only the box does not need to be IP65 class (waterproof), but if it's to be used out of doors IP65 is a must. Also, all relevant precautions in respect of electrical safety must be observed when installing the board in a box and fitting the wiring.

(080395)

Inside the TMR0 interrupt macro, when there is a TMR0 interrupt the software checks if it was a zero-crossing overflow or an TMR0 timer overflow interrupt. If a zero crossing interrupt occurred the TMR0 register is preset with the DMX value after calculation, and TMR0 is started again this time as a timer. The next interrupt-on-overflow effectively marks the triac firing, this is done with three

Internet Links and References

[1] Portable DMX512 Tester, Elektor June 2002. www.elektor.com/010203

[2] www.elektor.com/080395

[3] www.elektor.com/e-blocks

About the author

Per Stegelmann (44) lives in Denmark. Trained as an electronics technician, his current profession is building and repairing test equipment for the production of hearing aids. Per writes: "*When one of my colleagues asked me if I had ever seen a program for developing microcontroller code that used icons to code with, he showed me a demo version of Flowcode, the software tool for E-Blocks. I tried it out, and soon I was hooked — it was a whole new way to do the code development. Earlier I hade done some assembly language and C coding, but this I had never tried before. I had seen some ads for Flowcode but thought it would not be constructive or capable of producing useful code. As it turned out, Flowcode proved very useful indeed producing valuable code while also making the documenting of your program a breeze — you just print out the flowchart. So I bought a licence and an E-Blocks PIC Multiprogrammer, and a few weeks later I had my first application up and running.*"

Mobile, Text, CallerID

ATM18 will never work alone again…

YOU'VE GOT MAIL!

By Grégory Ester (France)

'Remote control', two little words that very often form a catchy headline — and rightly so. We always seek to dominate machines, control the situation — we like to feel all-powerful. With this project, all you have to do is either identify yourself through the caller ID displayed to the ATM18 or go right ahead and send a text to your favourite board for it to take the appropriate action.

Beauty (ATM18) and the Beast (GM862-GPS)

Telit's GM862-GPS [1] is ideally designed for building remote control applications. The module includes a 20-channel SIRF III GPS along with a 4-band GSM/GP(R)S modem, with a place to insert a SIM card. An antenna has to be added for the GPS part (note that the GPS function is not being used here) and a second one for the GSM. A test or evaluation board specially designed to take this module is available over the Internet [2].

For processing the data from the GM682-GPS module, we're going to be using ATM18 [3] and its faithful two-wire LCD display [4], along with the relay board [5] belonging to the same family. And to quickly add a signalling function to our board, we're going to use our familiar *BlinkM MaxM* giant RGB pixel with I²C drive, which has already proved itself in the article 'Talk Show' [6].

Taming the Beast…

Before implementing the GM862-GPS module, solder the pin sockets into the
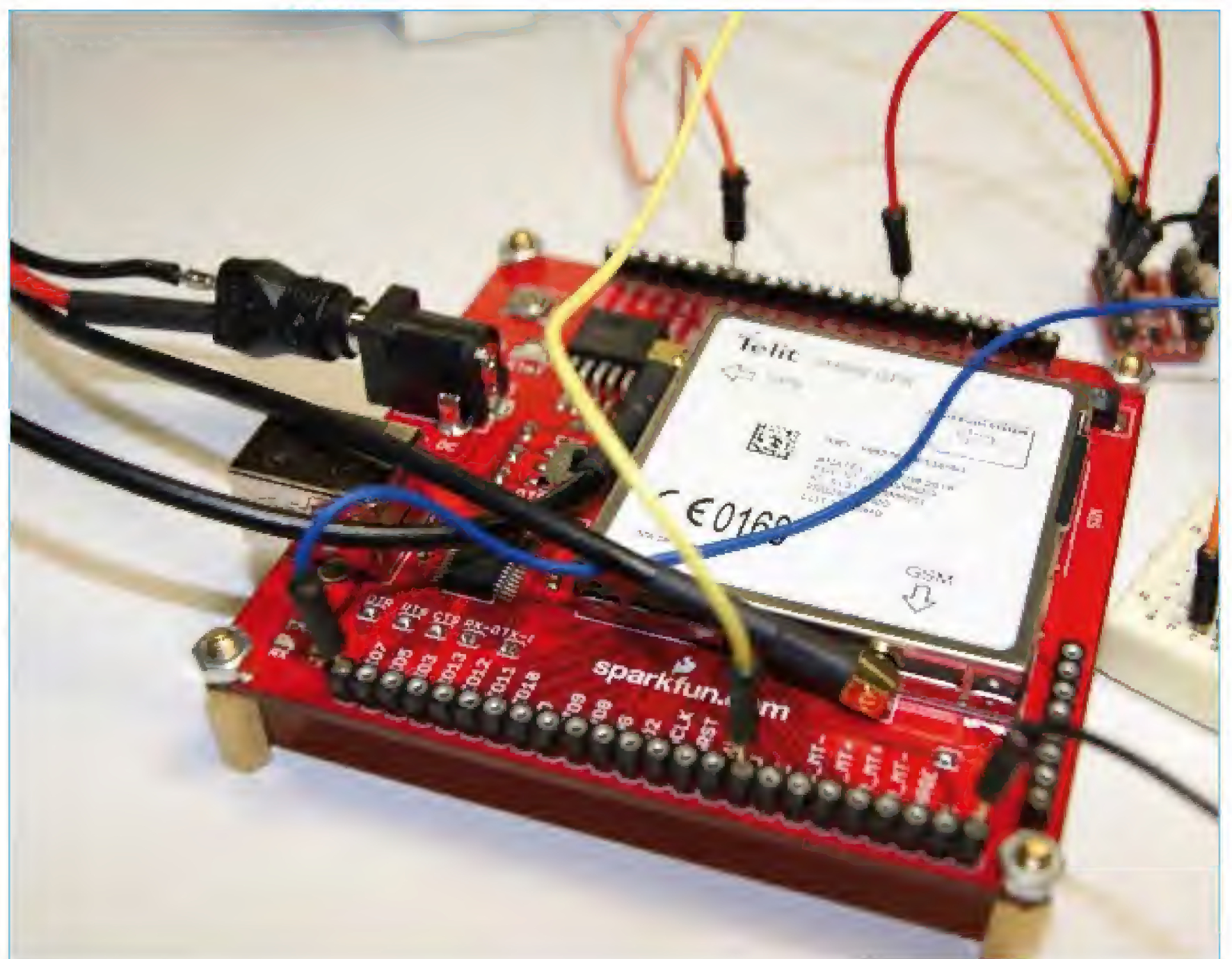
Figure 1. The GM862-GPS comfortably installed.

Elektor Products & Services

- Eight relay board: Elektor #071035-72
- Extension port board: Elektor #071035-95
- ATM18 controller board: Elektor #071035-91
- ATM18 piggy-back board: Elektor #071035-92
- Two-wire display: Elektor #071035-93
- Microcode (free download): # 110139-11.zip
  from www.elektor.com/110139
- Hyperlinks in article

Figure 2. Hercules is back!

Figure 3. A small handful of components to get sound.

small test board (**Figure 1**) which will make it easier for you to connect it up with the ATM18. Once you've done so, plug in the GPS antenna before inserting the GM862-GPS module into its socket, then connect up the GSM antenna. Set the DIL switch to DTR HI (GM862-GPS programming mode disabled).

Arm yourself with a USB A male to USB B male cable to connect the board to the PC. It may be necessary to install the 'FTDI' driver [7].

Turn on the modem by pressing the ON/OFF button for 1 s, just as you would do with your mobile. At this stage, the red POWER LED will be lit and the STATUS LED flashes at 1 s intervals indicating that the GM862-GPS is not registered on the network. Indeed, for the moment, no SIM card has been inserted.

Now the serious stuff can begin! The modem will respond to the commands sent over its serial link as long as the latter always start with the ASCII characters AT (abbreviation of ATtention) and end with a carriage return <CR>. In order to avoid breaking up this article with loads of 'AT' commands with comments, these have all been grouped together and indexed in **Table 1**. When a read or write command is sent, the reply is always visible as <CR><LF>OK<CR><LF> if the command is recognized, or <CR><LF>ERROR<CR><LF> if it is not (possibly with details of the error).

Let's start by testing the link we've just set up physically between our GM862-GPS and the computer, on which you will already have installed the well-known Hercules terminal [8], for the time being configured at 115,200 baud. The SIM card is not inserted for the moment. Before we send our first command, we're going to enable the display of special characters in their HEX form. To do this, position the mouse cursor within the active Received/Sent data window, one little right click will bring up a contextual menu. Select Special Chars > Hexadecimal.

Figure 2 shows that the link has been established properly. It's worth noting that the 01 command you've just sent is automatically repeated by the modem, ECHO is thus fol-

lowed by the expected response. To avoid cluttering up our table, we will no longer make special characters appear, hence command 02 is identical to command 03, the GM862-GPS module is indeed manufactured by Telit (commands 03 and 04). Command 06 tells you about the useful received signal strength, rssi = 1 corresponds to −111 dBm. The step is 2 dBm, consequently the value rssi = 11 corresponds to an attenuation of −91 dBm. The higher the RSSI, the better the level of reception.
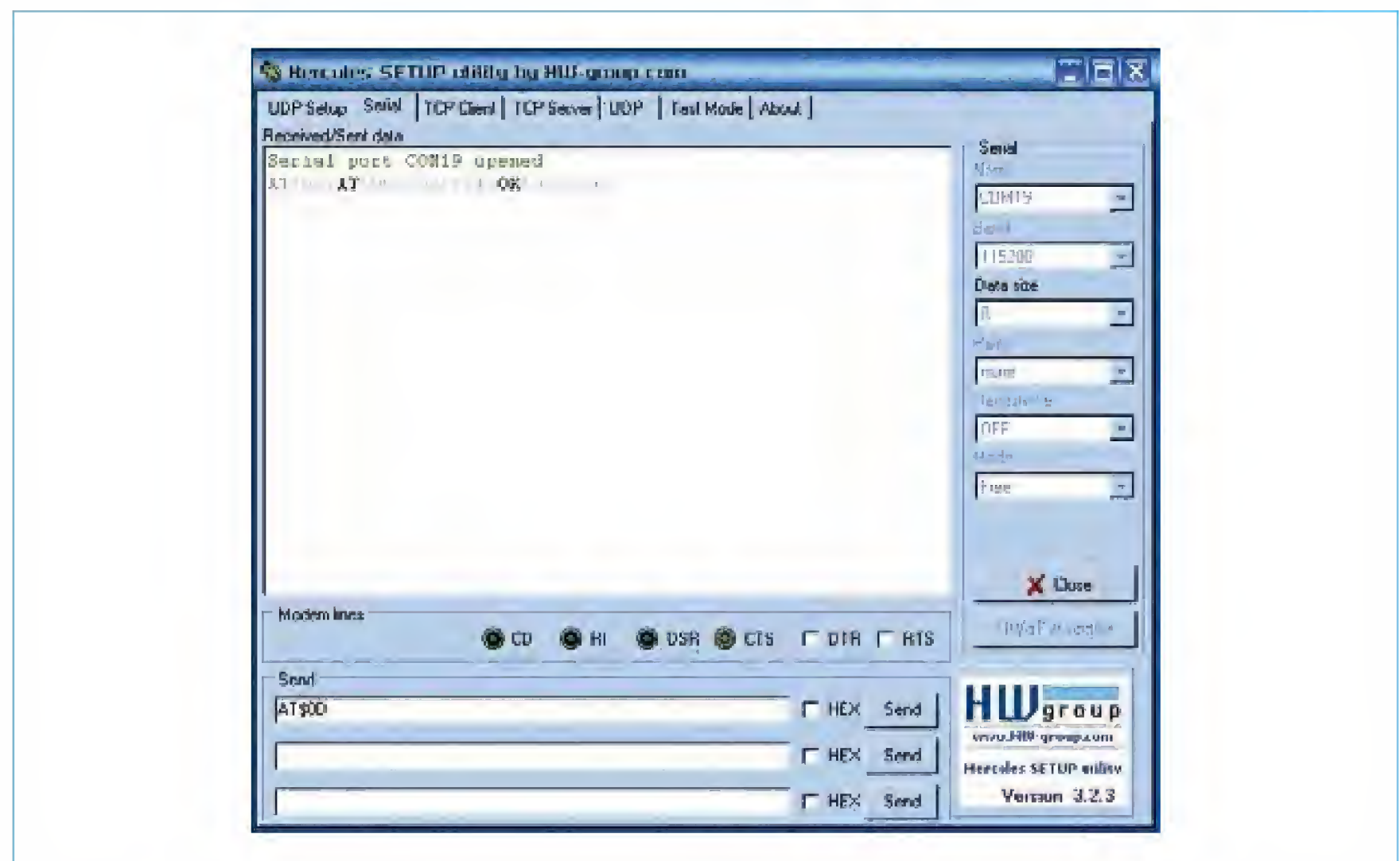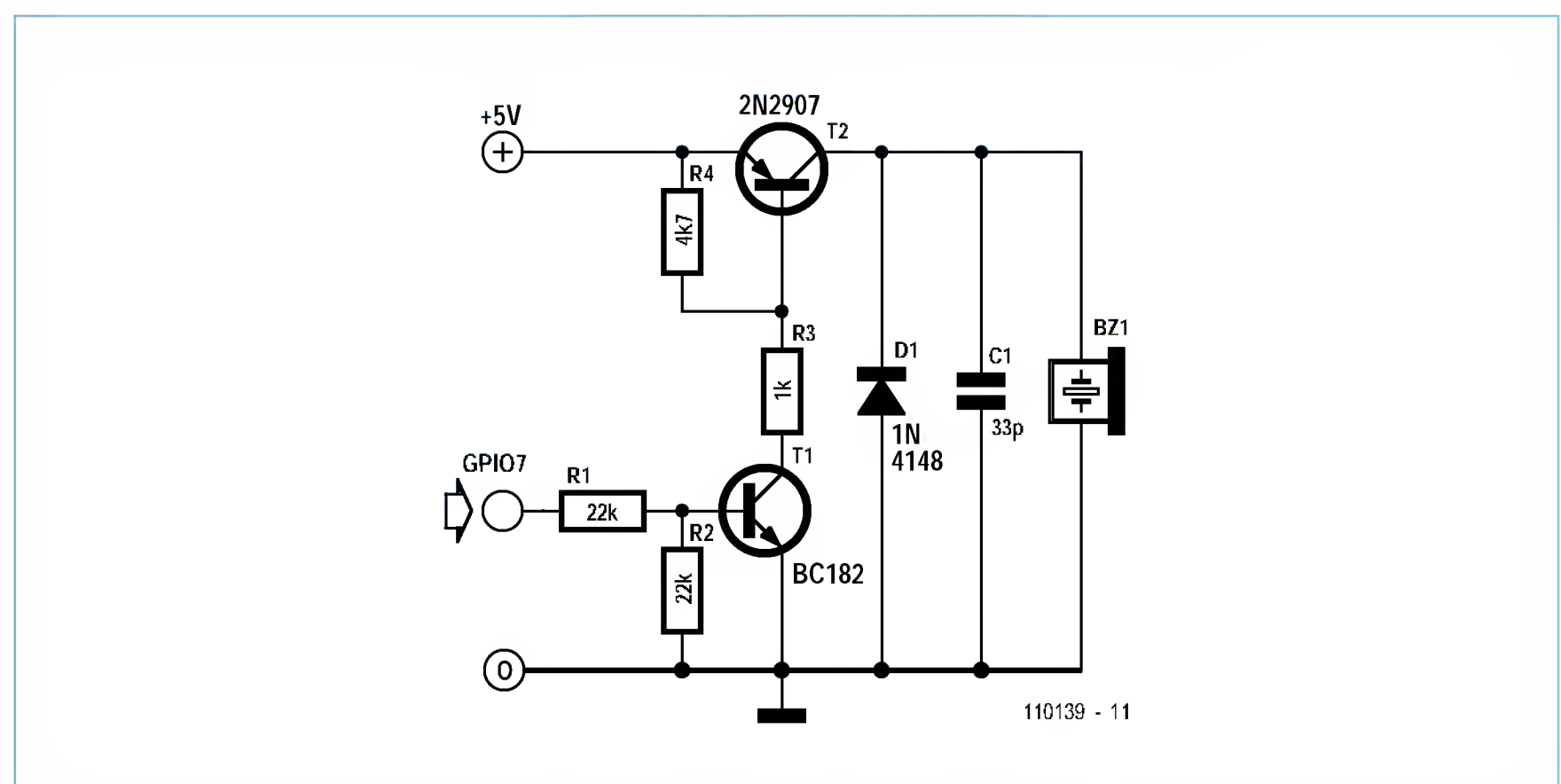
The Beast makes itself heard

The GM862-GPS has several general-purpose input/outputs. We're going to take a look at the pin named GPIO7, which can be configured as either an input or an output, but also in an 'ALTERNATE FUNCTION'

mode which, specifically for this pin, corresponds to the Buzzer Output mode. Command 07 allows you to enable pin GPIO7 in the above-mentioned mode. Command 08 lets you check the configuration of all the pins. All that now remains to be done is to tell the GM862-GPS module that sounds should now be sent to pin GPIO7 that has just been configured for this purpose. Command 09 is the one we need to do this.

A handful of components is necessary to adapt the signal from GPIO7 to the sounder (**Figure 3**). This little circuit can easily be built on prototyping board or breadboard. 5 V and GND will be connected to the 5 V rail on your ATM18 board.

Before introducing the ATM18 to the GM862-GPS, let's set the serial link transfer
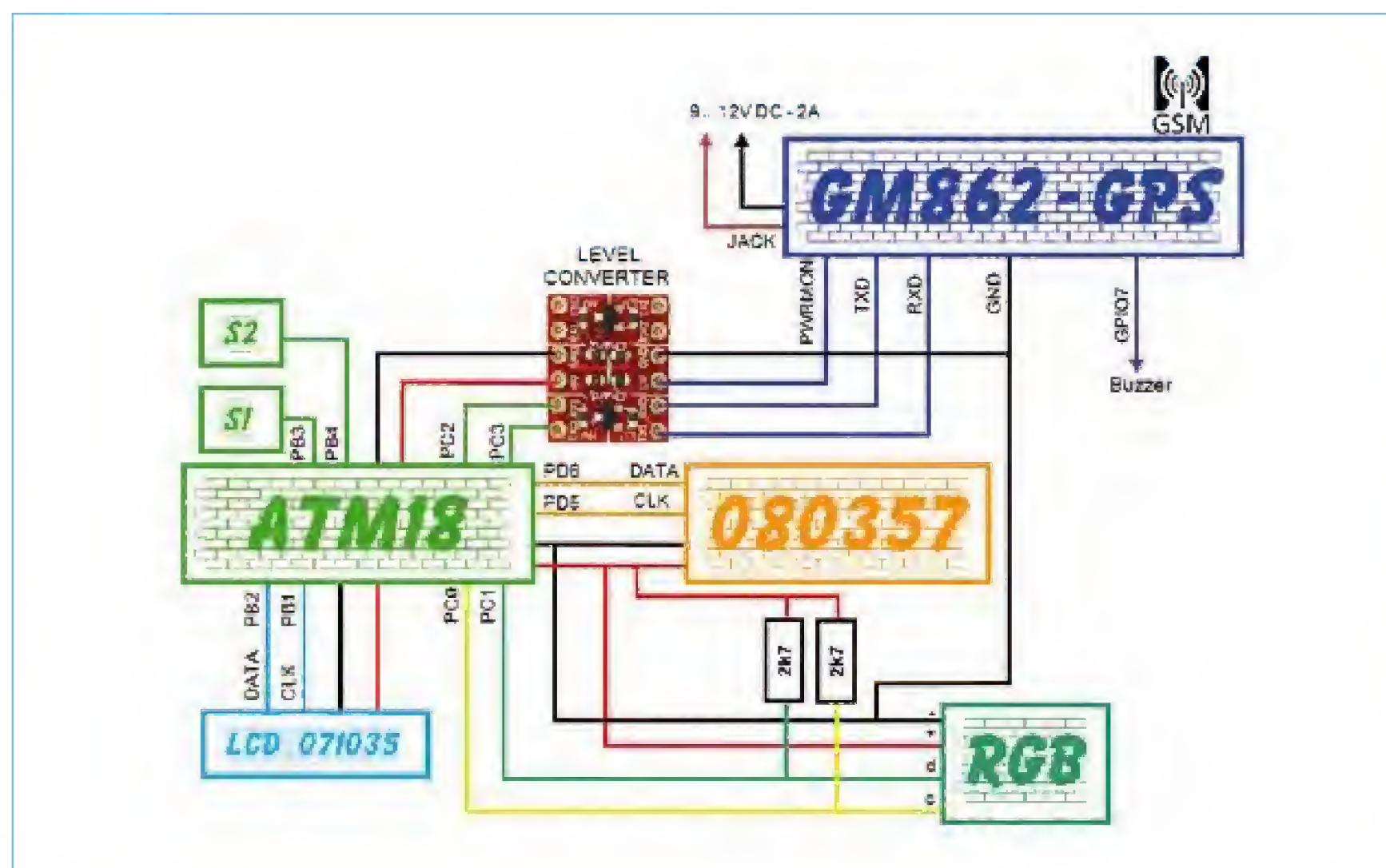
Figure 4. Wiring block diagram for the whole assembly.

Figure 5.

speed to 9,600 baud using command 12. Power down the module by unplugging the USB cable from the PC, then insert the SIM card. From this point on, it's preferable to power the modem board from its own 9–12 V, 2 A PSU. The board will still let you plug the USB cable back in so as to continue communicating with your modem via the Hercules terminal, and the external PSU will take precedence over the supply from the computer's USB port.

Before going any further, we need to use command 14 to enter the SIM card's PIN code. If the code is correct, when you confirm it you'll get a sound signal and if the GM862-GPS is registered on the network, the LED_STAT indicator will flash at 3 s intervals.

Execute commands 07 and 09 again in order to enable the ringing, then call your modem by dialling its number on another phone. You will hear it ring and the Status LED remains lit; at the same time, a message RING is sent to the terminal.
If you don't like the default ringtone, you can change it using command 15.

The Beast expresses itself
Without re-reading the above explanations, try now to configure pin GPIO7 aiming to enable it to receive incoming signal tones. The response returned by the module after executing command 18 tells us that the SIM card (SM) can be used to both read, write, send, and store text messages. Command 19 means that all read and write commands will be performed to the SIM card,

which has 50 spaces; three messages are currently stored in memory.
To send a text, you start by enabling the TEXT mode using command 20. When you send a text, it goes first to your message centre (SMSC), so this number needs to be present somewhere. This parameter is normally already entered and command 21 will return the answer.
Command 22: a message is sent in two steps. First, you need to fill in the recipient's number, then enter the message and confirm using 'Ctrl-Z' ($1A); the message is sent directly to the network without being stored on your SIM card.

When you receive a text, command 23 will be needed, so that the GM862-GPS (ME = Mobile Equipment) can signal the presence of a message, and above all its location, to the Hercules terminal (TE = Terminal Equipment). From now on, if the GM862-GPS receives a text, the terminal will receive a message formatted like this: +CMTI: "SM",5 which means you have received a new message and it is stored in position 5 on your SIM card. To read it, simply execute command 24. Right, you've read it? Now you can delete it using command 25.

Beauty and the Beast: first meeting
One of the two programs described below may be loaded into the memory of our ATM18's ATmega88 microcontroller. All the elements should be connected up as per the block diagram in **Figure 4**. The bridge
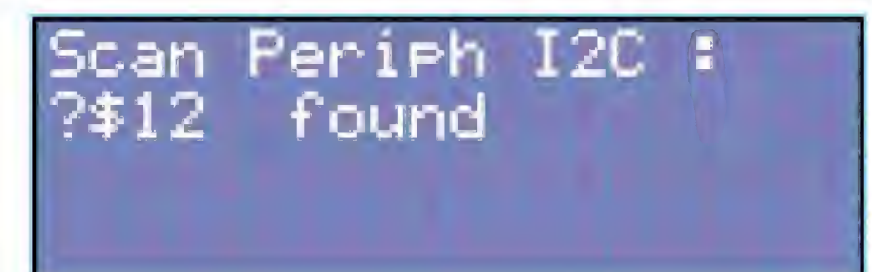
points between contacts RX-O and TX-I on the red test board carrying our modem must be open (two little touches with the soldering iron are all it takes to remove the solder bridges). The whole thing is powered up and the GM862-GPS modem is activated by a long press on the ON/OFF button.
The first program (110139-I_GM862-GPS_ ATM18_P1.bas [9]) lets you toggle the state of relay 1 by calling up your unit, as long as you are identified as the administrator. To define the administrator, all you have to do is modify the constants `Const Admin_ phone = "xxXXXXXXXX"` and `Const Code_pin = "XXXX"` by entering your phone number and PIN code. Of course, for you to be identified, you must have caller ID enabled so as to release your number to ATM18. Compile, load, and it's ready!
If the caller's number is recognised as that of the administrator, the BlinkM MaxM giant pixel will light up green; if not, it will turn red.

Once the program has been run, here's the sequence of what will happen:
Initially, once it has automatically recovered the I²C address of the BlinkM MaxM module (**Figure 5**), the program tests the peripherals. The RGB module lights up red and the eight relays are briefly energized.
Echo is disabled for the commands sent. The PIN code is entered, followed by a wait of a few seconds, then the link is tested and the GM862-GPS module is initialised (**Figure 6**). If the PIN code has already been entered at a previous start-up, the message in **Figure 7** will be displayed.

At this stage, pressing button S1 on the ATM18 board lets you move on to the next screen (**Figure 8**) so as to choose the type of ringtone (the first line tells you if your GM862-GPS is "picking up" the network). If you keep S2 pressed within 1.5 s of this

Figure 6.

Figure 7.

Figure 8.

Figure 9.

Figure 10.

Figure 11.

screen's appearing, the 32 ringtones are played briefly at 1.5 s intervals. Release S2 when you hear the one you want. If you keep S2 pressed for more than 1.5 s, there will be a delay of 8 s before moving on to the next ringtone, allowing you to listen to the whole of the ringtone. Releasing S2 lets you move on to the next step (**Figure 10**). As soon as the screen shown in **Figure 11** appears, you'll at last be able to call up ATM18!

From the phone registered as the administrator, dial the number corresponding to the SIM card inserted in the GM862-GPS; your chosen ringtone will sound, relay 1 is energized, and the LED lights up green (**Figure 12**). Wait one minute in order to be able to make another call. Now call your circuit from a phone with a different number. The message in **Figure 13** appears, and this time the LED lights up red!

Sending text messages to the ATM18

The preceding project has the advantage of costing you absolutely nothing in call charges. Moreover, it's perfectly possible and very easy to extend the program. Since we know that on each call the string RING is sent to the ATM18, we could envisage counting the number of rings as a condition for activating one particular device (relays, in our case). As a second development, we could also envisage adapting the program for hearing-impaired people so as to assign a different, unique colour to the RGB pixel depending on who is calling. Upon arriving, visitors would no longer need to use the door intercom — once at the entrance to the block of flats, all they'd have to do would be call the unit in the hearing-impaired person's flat, and their presence would be indicated by a colour that had been pro-

grammed specifically for them.
The second program (110139-I_GM862-GPS_ATM18_P2.bas [9]) goes a bit further, since it will enable you to send a text to your unit in order to drive all eight relays at once, assigning each of them either a logic '0' (not energized) or '1' (energized). What's more, you'll get a receipt of delivery to confirm that your request has been carried out. It's worth noting that you can also merely interrogate the unit to find out the logic state of all the relays without actually activating any of them. But how do I do this? And how does it work?

At power-on, operation is identical to the previous circuit, echo is disabled for the commands sent, the PIN code is entered, followed by a wait of a few seconds, then the link is tested and the GM862-GPS module is initialised. So you get a display like those shown in **Figures 6 or 7**. Pressing S1

Figure 12. Call received from the Administrator.
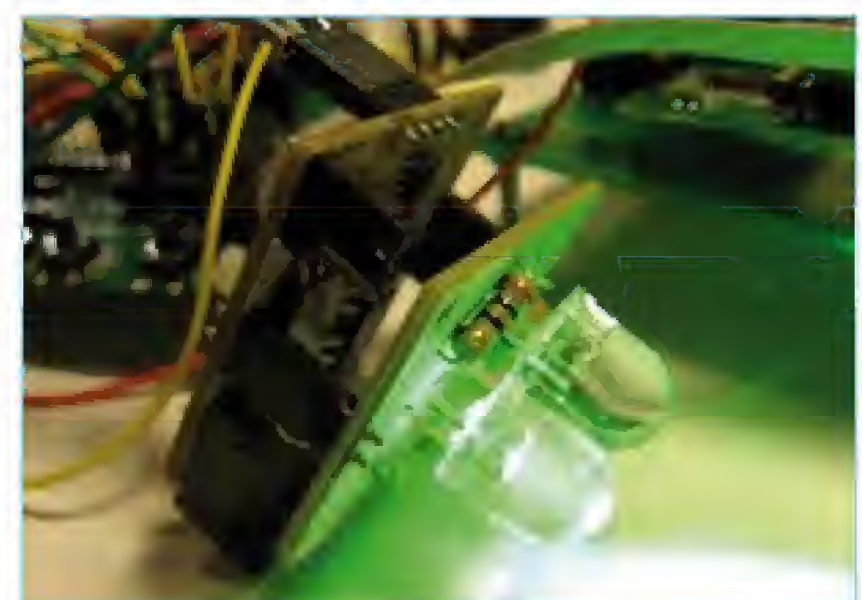
Table 1. The labours of Hercules			
Index	Command	Expected response	Meaning of response
	`AT+CGMR$0D`	`07.02.403 OK`	Software Version
	`AT$$GPSSW?$0D`	`$GPSSW: GSW3.5.0Telit_3.5.00.00-C23P2.00a OK`	GPS Software Version
	`AT##SELINT?$0D`	`#SELINT: 2 OK`	AT Interface Style 2
01	`AT$0d`	`{0D}{0A}OK{0D}{0A}`	Link established
02	`AT+CGMI$0D`	`AT+CGMI{0D}{0D}{0A}Telit{0D}{0A}{0D}{0A}OK{0D}{0A}`	Manufacturer: Telit
03	`AT+CGMI$0D`	`Telit OK`	Manufacturer: Telit
04	`AT+CGMM$0D`	`GM862-GPS OK`	Model: GM862-GPS
05	`AT+CGSN$0D`	`358278006334593 OK`	IMEI code: 358278.00.633459.3
06	`AT+CSQ$0D`	`+CSQ: 11,0 OK`	+CSQ: rssi = 11, error rate less than 0.2%
07	`AT##GPIO=7,0,2$0D`	`OK`	Pin 7 is configured in Buzzer Output mode
08	`AT##GPIO?$0D`	Displays the logic states of all the Input/Outputs	
09	`AT##SRP=3$0D`	`OK`	Ring output to sounder
10	`AT##SRP?$0D`	`#SRP: 3 OK`	Ring output configured in mode 3
11	`AT$$GPSSAV$0D`	`OK`	Saves GPS parameters
12	`AT+IPR=9600$0D`	`OK`	Serial port set to 9600 baud
13	`AT##SHDN$0D`	`OK` with an audible signal	Modem disconnected from network and then powered down
14	`AT+CPIN=7453$0D`	`OK`	PIN code is entered
15	`AT##SRS=7,10$0D` et `AT##SRS=0,0$0D`	`OK` `OK`	Plays the 7th of the 32 tunes for 10 s (max. = 60 s) and saves this new ringtone as the default ringtone
16	`AT+CCLK="11/02/04, 20:48:00+04"$0D`	`OK`	Sets time and date: 20h48m00s 4Feb2011 GMT+1
17	`AT+CCLK?$0D`	`+CCLK: "11/02/04,22:06:22" OK`	Reads date and time
18	`AT+CPMS=?$0D`	`+CPMS: ("ME","SM"), ("SM"), ("SM") OK`	Storage memories available?
19	`AT+CPMS?$0D`	`+CPMS: "SM",3,50,"SM",3,50,"SM",3,50 OK`	Current message storage preferences
20	`AT+CMGF=1$0D`	`OK`	Activates TEXT mode for sending text messages
21	`AT+CSCA?$0D`	`+CSCA: "+33609001390",145 OK`	SMSC message centre number
22	`AT+CMGS=0682834725$0D HELLO BB$1A`	`> _` `+CMGS: 3 OK`	Enter your text message The text is sent
23	`AT+CNMI=1,1$0D`	`OK`	Configuration possible for receiving a text message
24	`AT+CMGR=5$0D`	`+CMGR: "REC READ","+33682834725","","11/02/05,21:57:49+04" Hello from BB OK`	The message stored in location #5 is read
25	`AT+CMGD=5$0D`	`OK`	The message situated at location #5 is deleted
26	`AT+CMGR=5$0D`	`+CMS ERROR: 321`	Invalid Memory Index, there is no (or no longer) a message to be read in this location.

Figure 13. Call received from an unknown number.
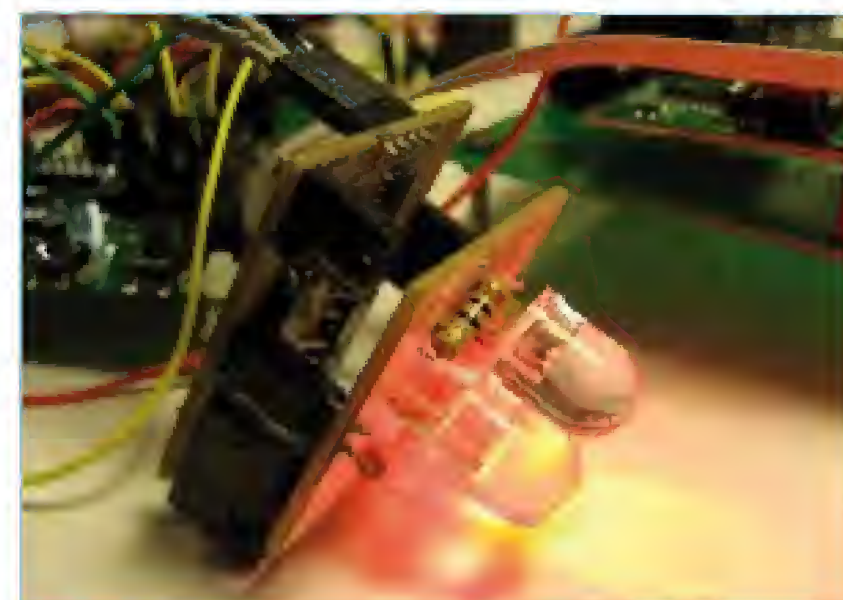
takes you on to the next screen (**Figure 14**). The network status is verified and the second line is displayed, as long of course as you are registered on said network. ATM18 and GM862-GPS are standing by for your orders!

Now without waiting any longer, grab your mobile — the same one as just now — and text "00001111" to your circuit. Didly-dee! Did you hear that? The GM862-GPS reacts! At the same time, a message is added to the LCD screen to tell us that a text has been received and stored in memory zone #1 (**Figure 15**). The contents of the text you have sent is displayed on the fourth line and relays RE1 to RE4 are energised. The receipt is then sent, indicated by the message REPORT SENT! on the fourth line, and on your phone you receive a text to show that the action has been successful. And not

even 30 s have passed!
Now send ATM18 the following message, or rather, character: "R". This time, on the fourth line of the LCD screen, the message REPORT? is displayed (**Figure 16**) indicating

that you want only to receive the logic state of all the relays, but no action is required. Naturally you get a text message back with the reply to your request (**Figure 17**).

(110139)

Internet links

[1] www.telit.com/en/products/gsm-gprs.php?p_ac=show&p=7

[2] www.sparkfun.com/products/281

[3] www.elektor.com/atm18

[4] www.elektor.com/071035

[5] www.elektor.com/080357

[6] www.elektor.com/100360

[7] www.ftdichip.com/FTDrivers.htm

[8] www.hw-group.com/products/hercules/index_en.html

[9] www.elektor.com/110139
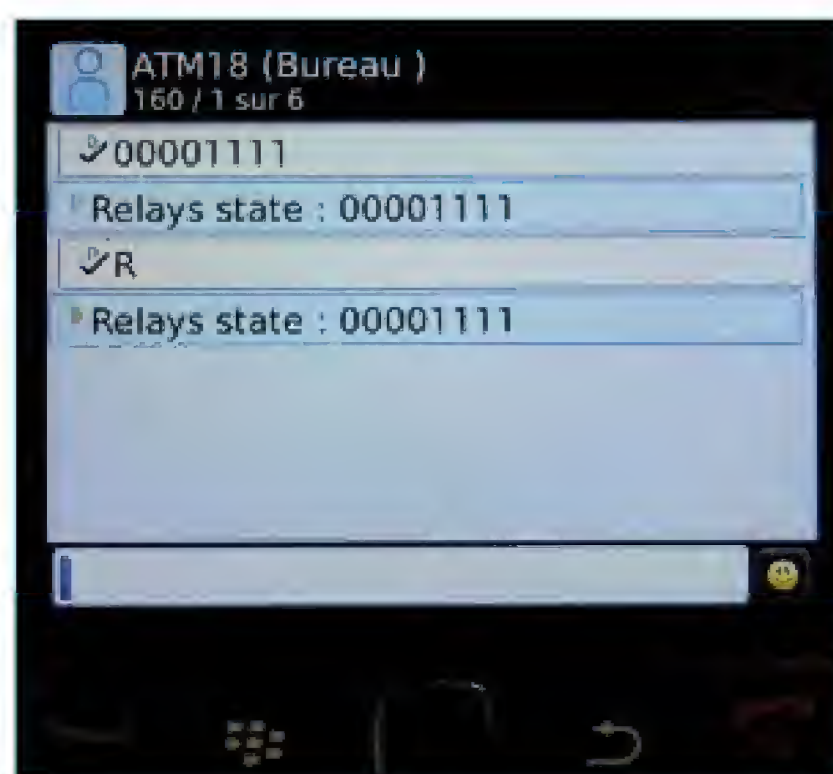
Figure 14.

Figure 15.

Figure 18. The phone display, showing the exchanges that have just taken place.

Figure 16.

Figure 17.

Wave Sound Generator
Restful electronics

The sound of the sea, or more precisely the sound of waves, has a relaxing and restful effect on people. In these busy and hectic times there are many people who need this, for example to help them fall asleep at night. It's fairly easy to simulate the sound of waves with a compact bit of electronics.

By Harry Baggen (Elektor Netherlands Editorial)

Elektor Products & Services

- PCB: # 100922-1
- Fully assembled and tested board: # 100922-91
- Special enclosure available
- PCB layout (free download): # 100922-1.zip
- All items accessible via www.elektor.com/100922

In some way or another, the sound of waves breaking on the shore has a calming effect on most people. The reason for this has been investigated by innumerable scientists, but up to now nobody has been able to find a definitive explanation for this phenomenon. Of course, the reason isn't all that important; the fact is that many people benefit from listening to wave sounds. Especially for people who suffer from insomnia, wave sounds can provide a good alternative to sleeping pills. Even if you are not an insomniac, it's very pleasant to go to sleep at night with the sound of gently rising and falling waves in the background. You're virtually guaranteed to wake up the next morning feeling especially well rested.

It doesn't take a lot of electronics to produce simulated wave sounds. A variety of circuits for this have been published in past issues of Elektor magazine. In light of recent growing interest in a design for a wave sound generator, we took a close look at the previous version (published in February 1996) and transformed it into an updated and very compact design.

Noise, square waves and triangle waves

When you think about wave sounds, it's natural to immediately think of noise. That's something we normally try to avoid as much as possible in electronic circuits, but here we expressly want to produce noise. There's a well known way to do this, which is to reverse-bias the base-emitter junction of a transistor so that it acts like a zener diode. In the schematic diagram shown in **Figure 1**, this task is handled by T2. Resistor R15 limits the reverse current through the junction, resulting in a voltage across the junction of around 7 to 9 V. The noise signal is feed via coupling capacitor C7 to a gain stage built around T1, which amplifies the signal considerably. The gain can be adjusted by potentiometer P2. The amplified signal then goes via C10 to a small semi-discrete power amplifier composed of IC2a, T3, T4, and a handful of other components. Thanks to the push-pull output stage formed by the discrete transistors, this mini-amplifier can deliver enough current to drive a small loudspeaker directly with sufficient power to produced clearly audible noise in a living room or bedroom.

So now we can listen to noise, but that's not so special. The noise still needs to be modulated to produce the characteristic sound
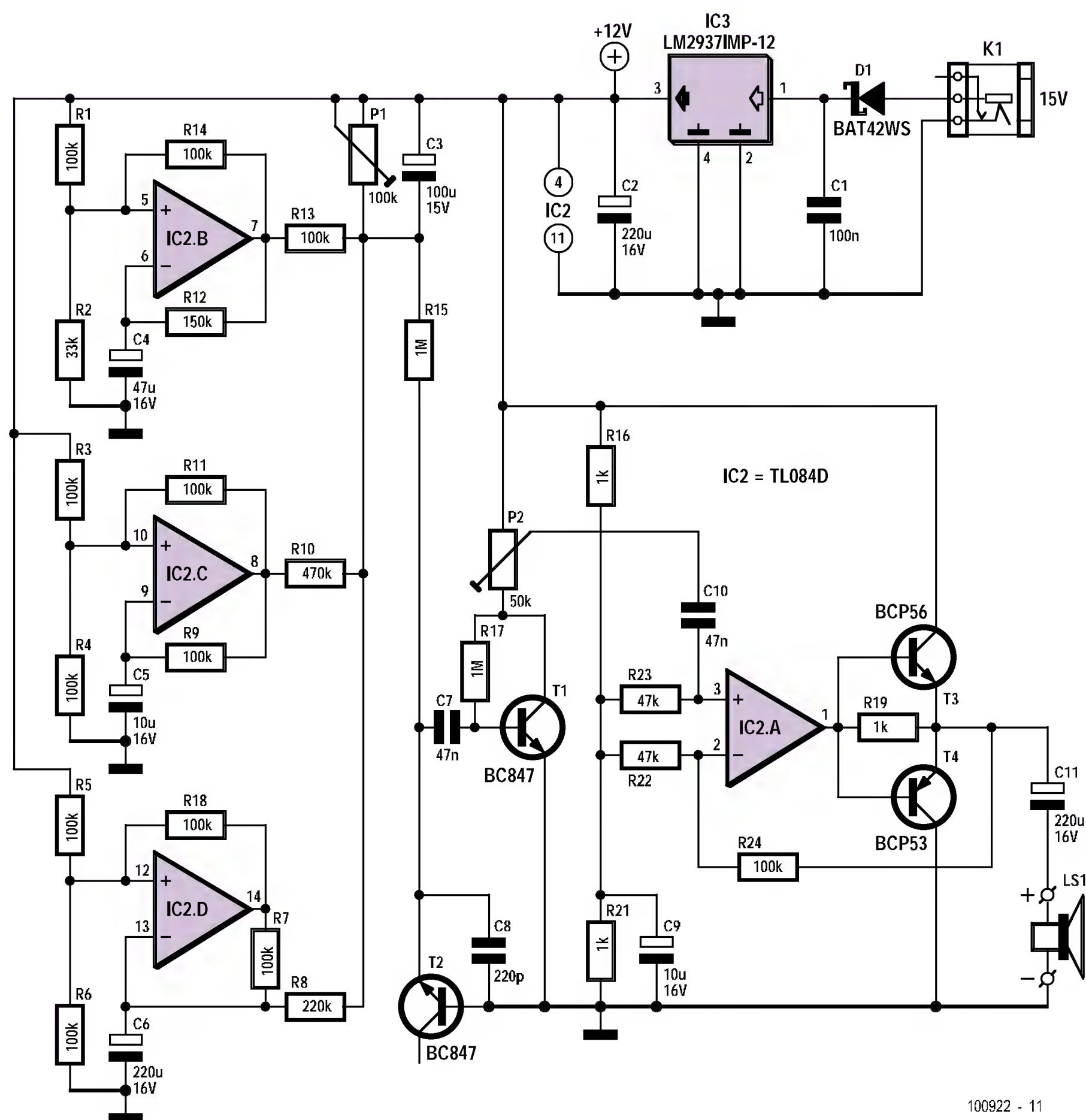
Figure 1. The wave sound generator circuit consists of three square-wave generators, a noise source, a gain stage and a small output amplifier.

of breaking waves. For this purpose, we use the three remaining opamps in the same package as IC2a. Each of these opamps (IC2b, IC2c and IC2d) is wired as a square-wave generator, with widely different frequencies determined by the time-constant networks R12/C4, R9/C5 and R7/C6. The upper two oscillator circuits (built around IC2b and IC2c) generate square-wave signals that are fed via R13 and R10 to C3. This capacitor integrates the step changes in the signal levels to produce a sort of triangle-wave signal. This triangle-wave signal causes the current through T2, and with it the level of the generated noise signal, to vary periodically in simulation of the sound of rising and falling waves. The bottom oscillator produces such a low frequency that C3 has scarcely any integrating effect. For this reason, the signal at the junction of

R7 and C6, which already has a triangular waveform, is connected via R8 to C3 instead of the square-wave signal at the output of the oscillator. As a consequence of C6 being connected to the positive supply voltage via R8 and P1, the voltage across capacitor C6 is closer to a sawtooth than a triangle wave. This signal simulates the effect of the large waves that occasionally roll onto the beach.

As the three oscillators operate at unrelated frequencies due to the values of the timing components and are not synchronised to each other, the resulting waveform is fairly random. Here are a few numbers for your information: the period of the square-wave signal from IC2b is nearly 10 s, the period of the signal from IC2c is approximately 1.5 s, and the period of the slowest waveform (from IC2d) is almost 1 minute. Summing

these three signals yields a slowly varying voltage at the junction of P1 and R15 with a level between 9 V and 11 V.

The frequency spectrum of real waves is actually much broader, with very low frequencies as well as very high frequencies, but in light of the fact that the frequency range of the miniature speaker used here is fairly limited, we didn't try to simulate the full spectrum.

Low-drop voltage regulator IC3 ensures that the entire circuit is powered from a stable 12 V supply voltage. Diode D1 is included in the circuit to provide reverse polarity protection, so that nothing will go up in smoke if you accidentally connect the AC power adapter's output the wrong way round. The operating input voltage range is 15–25 V, and the maximum current consumption is 100 mA.

COMPONENT LIST

Resistors (SMD)
R1,R3,R4–R7,R9,R11,R13,R14,R18,R24 =
    100kΩ (0805)
R2 = 33kΩ (0805)
R8 = 220kΩ (0805)
R10 = 470kΩ (0805)
R12 = 150kΩ (0805)
R15,R17 = 1MΩ (0805)
R16,R19,R21 = 1kΩ (0805)
R22,R23 = 47kΩ (0805)
P1 = 100kΩ trimpot (3306W)
P2 = 50kΩ trimpot (3306W)

Capacitors (SMD)
C1 = 100nF (0805)
C2,C3 = 100µF 16V (case-d)
C4 = 47µF 16V (case-d)
C5,C9 = 10µF 16V (case-b)
C6,C11 = 220µF 16V (case-e)
C7,C10 = 47nF (0805)
C8 = 220pF

Semiconductors (SMD)
D1 = BAT42W
T1,T2 = BC847
T4 = BCP53

T3 = BCP56
IC3 = LM2937IMP-12/NOPB
IC2 = TL084ACD

Miscellaneous
K1 = power adaptor socket (CUI PJ-007)

LS1 = miniature loudspeaker, e.g. Kingstate
    KDMG20008, Farnell # 1502730)
PCB # 100922-1
Assembled board # 100922-91
    (see www.elektor.com/100922)
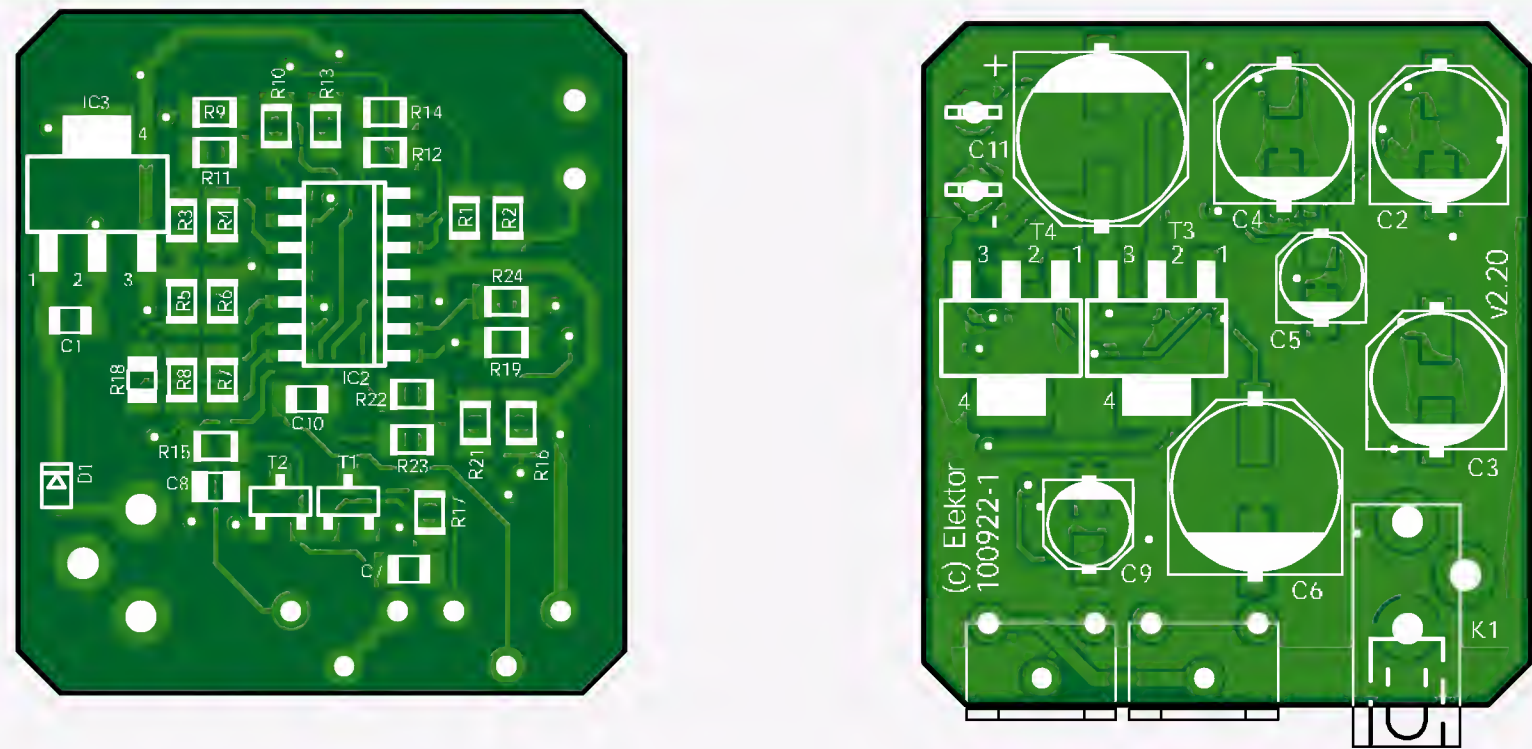
Figure 2. The PCB for the wave sound generator (shown here at 1.5 times actual size) is extremely compact thanks to the use of SMDs.

Compact PCB

To keep the circuit as small as possible, which makes it easier to fit it into an enclosure, we chose SMDs for all of the components. This allows the dimensions of the PCB (see **Figure 2**) to be limited to approximately 28 ⨯ 32 mm. Although this makes manual assembly a good deal more difficult, this should not form a problem for most potential users, since we also offer a fully assembled version of the circuit at a quite modest price. If you nevertheless wish to do things the hard way, you can download the PCB layout file free of charge from the Elektor website or order a bare PCB.

As already mentioned, thanks to its small size the board can be fitted in virtually any available enclosure. The necessary dimensions are largely dependent on the size of the speaker that is used. Especially for this project industrial designer Rein van der Mast has developed an enclosure in the form of a seashell, which interested DIYers can order. A link to his product will be posted on the project page for this article. The case yields an attrac-tive unit that can claim a fixed place in your home, such as on a bedside table.

It's a good idea to test and adjust the assembled board before you fit it into the enclosure. To do so, connect a loudspeaker and a suitable AC power adapter (15 V at 100 mA is sufficient) to the board. Start by setting both trimpots to the centre of their travel, and listen for noise coming from the speaker. P2 is a sort of volume control; you can adjust it to make the noise as loud (or as quiet) as you wish. Then adjust P1 to obtain an audible rising and falling effect in the noise level. You should still hear a certain amount of noise even in the 'quiet' passages. Once everything is working the way it should, you can fit the board in the enclosure and start enjoying wave sounds.

But please don't go to sleep right away — you still have the rest of the magazine to read!

(100922-I)

Figure 3. Artist's impression of the seashell enclosure specially designed by Rein van der Mast of SOLide for the Wave Sound Generator.

Reference

Wave Sound Generator, Elektor February 1996

The Finishing Touch
Design professional front panels for free

The first thing that catches the eye of the just-completed amplifier, measuring instrument or other electronic device is without doubt its 'user interface': the front panel.

Making an attractive (that's: professionally finished) front panel for an electronic circuit is a difficult hurdle to surmount for many electronics enthusiasts. In addition to a degree of materials knowledge, you also have to possess cutting, drilling and milling machinery and especially the skills to use them properly.

By Giel Dols (Elektor DTP/Graphics department)

Schaeffer AG, a German manufacturer of enclosures and front panels offers an efficient solution for this problem. With the aid of their software tool called Front Panel Designer, which you can download for free, anyone can now design professional front panels and have them manufactured.

The software is available in English, German and French and there are versions available for Windows, Linux and Mac OS [1].

In this article we show in a broad outline how easy it is to design a professionally looking front panel with Front Panel Designer and have it manufactured. The software has a very good help facility built in which gives an excellent explanation in every situation. So download the software and install it.

As an example, here we create a design for a front panel to be used in a 19" rack, to house the preamp [2] that makes it possible to connect an electric guitar to the Elektor Multi-Effects Unit [3].

Step 1: The characteristics of the front panel

The menu option *File -> New* opens a window where all the physical characteristics of the front panel have to be defined: basic shape, length, width and thickness. A special option here is the choice for front panels intended for use in a 19" rack, where the height is indicated in 'units'.

We obviously also need to indicate what material the front panel needs to be made of and (eventually) what colour it will be. Here we have the choice between aluminium (in various finishes) and Perspex.

The final option that can be defined is the radius for rounding the corners of the front panel.
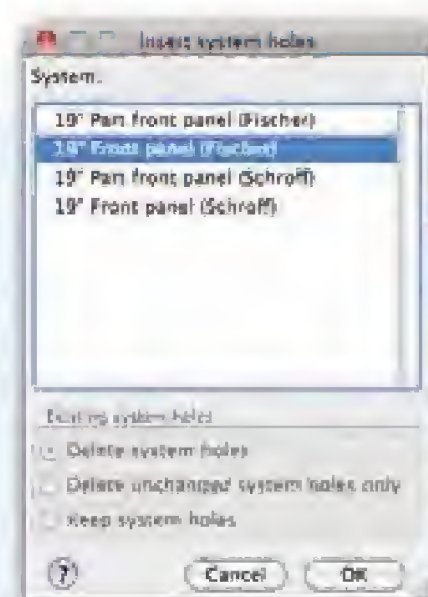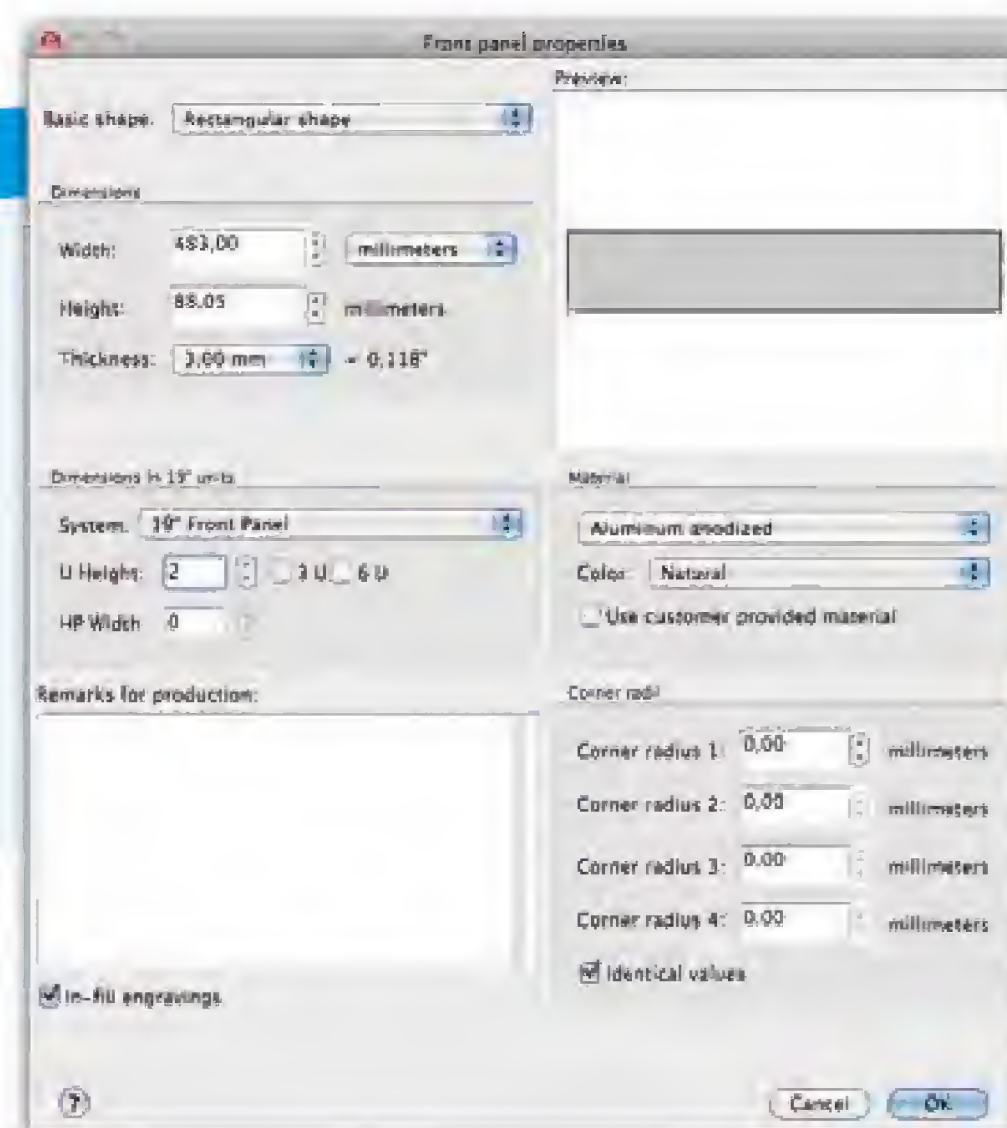
Step 2: Holes, holes and holes

This is definitely well thought through: a 19" front panel needs mounting holes in exactly the right places. The software takes care of all this. There are ready-made definitions for the mounting holes for two large manufacturers of 19" racks. Via *Insert -> System holes* you arrive at the menu where you can select the appropriate option.

The control panel for the Elektor Multi-Effects Unit requires an opening for the LC-display, five holes for the spindles of K1 and S1 through to S4. Another four holes are required to allow the PCB to be mounted behind the front panel.

Our preamp requires five holes: three for the potentiometers JP3, JP4 and JP5 and two for the 6.3-mm jack sockets

for the connections to the guitar and the Mute footswitch.

Select *Insert -> Drill hole* and click on the front panel where the hole has to be. Now there appears the drill hole menu. Here we define what the desired drill hole is supposed to look like. We chose a 3 mm hole intended for M3 socket cap screws with countersunk heads *(countersunk Allen screws)*. The software fills in all the relevant data and displays a window showing it will look like in cross-section.

When in *Select mode* and clicking on a part of the front panel, all the characteristics for that part then appear in the top left corner of the screen. Any of the characteristics can be changed in this window. So here we can enter very accurately the x and y coordinates of a part (in this case our drill hole).

These coordinates are relative with respect to the location of the origin. By default this is at the bottom left corner of the front panel. To avoid having to calculate the coordinates of every part on the front panel relative to this point, it is possible to move this datum point somewhere else. To do this, we click on Set origin and then click very accurately (after zooming in considerably) on the new place where we would like the origin to be. By positioning the origin at our drill hole it becomes much easier to put the other drill holes and the LCD display opening in their correct places.
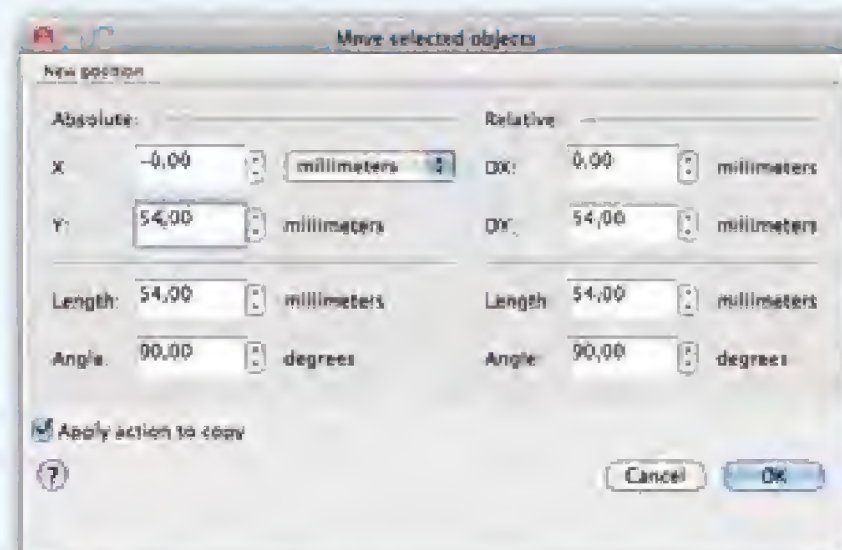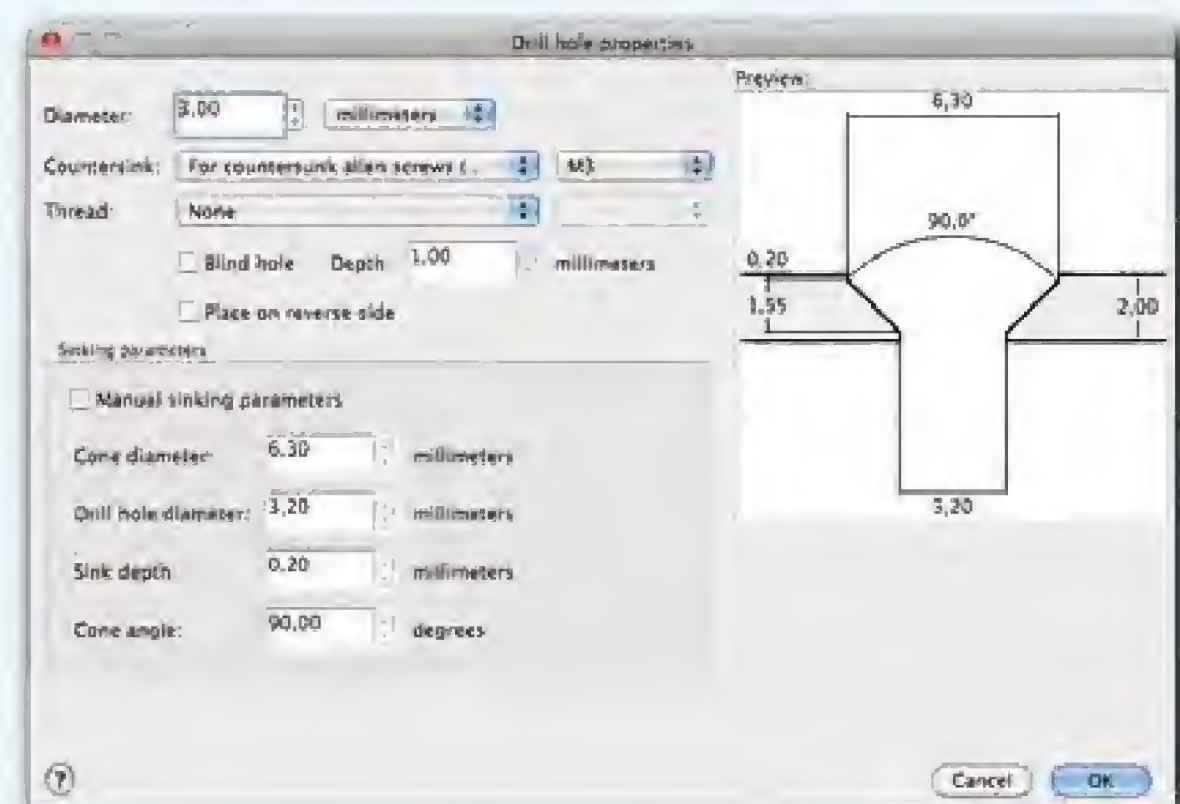
Positioning the other drill holes can also be done by using the move option. First select the drill hole already placed and then indicate via *Selection -> Move* the displacement for the second drill hole (in our case 54 mm). Don't forget to tick the box '*Apply action to copy*', otherwise the selected hole will be moved and not copied! Now select both drill holes and repeat the procedure; but this time with a horizontal displacement of 85 mm.

Now we have to draw the opening for the LCD display. Via *Insert -> Rectangular cutout* we arrive at a menu for entering the characteristics of our 'window': length, width and corner radius in our case. Select the window and enter in the '*Object properties*'-window (top left) the exact coordinates for the centre point of the desired display window.

Now it is time to position the holes for the four operating buttons S1 through to S4, followed by the spindle for K1. The y-coordinates for all these holes are the same (they are all on one line). We determine the x-coordinates by taking measurements from the original circuit board.

The software already has a modest, but handy, library with special openings for fans, switches, sub-D and SCART connectors from a number of well-known manufacturers. These special items are accessible via *Insert -> Macro object*. When an item is clicked, a preview appears in a window below the macro library. The software also offers the ability to select items or combinations of items yourself and store them an a user-library. To do this select *Selection -> Create macro*
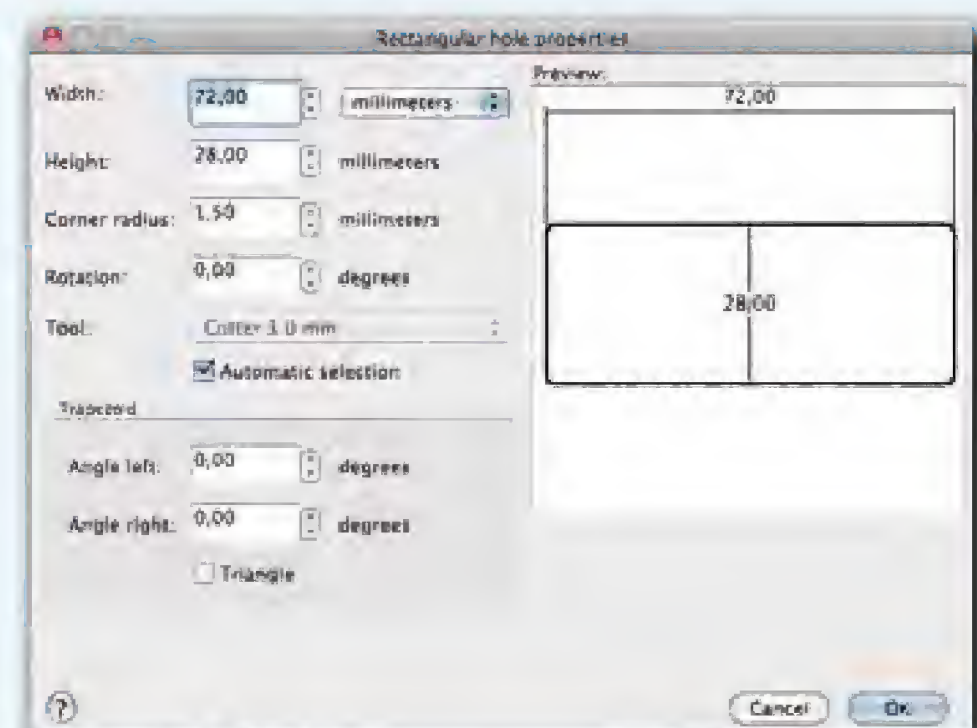
Step 3: Text

The menu option *Insert -> Text engraving* opens a window where you can enter the desired text and its characteristics. Each separately defined block of text can be selected at any time to position it accurately or to change it.

Step 4: What does it cost?

A click on the rightmost icon, clearly identifiable by two coins, opens a window with the price for the front panel.

The costs are specified to the smallest detail and there is also an overview with quantity pricing. Fortunately there is no room for surprises!
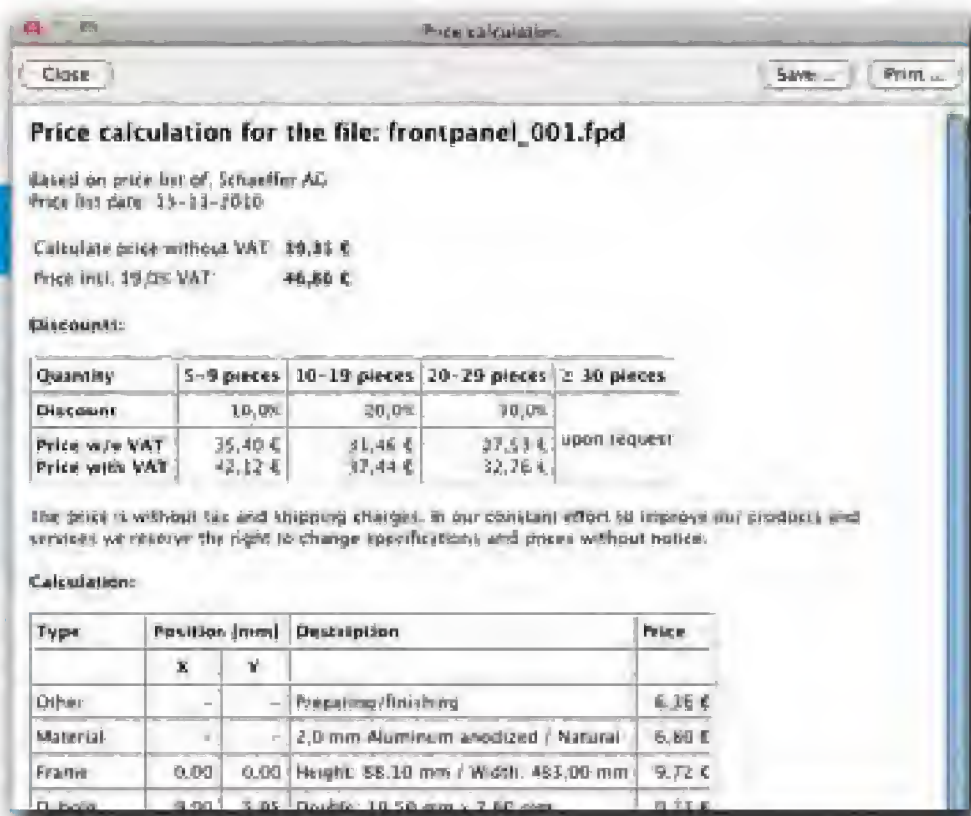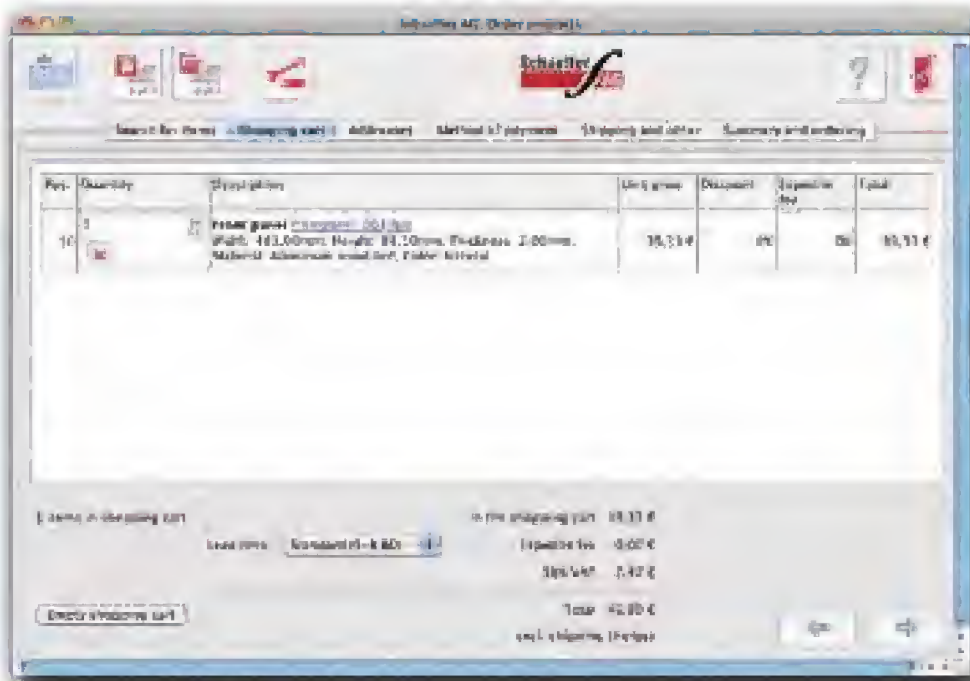
Step 5: Go ahead and order...

The final step before we can attach our newly designed front panel to our device is obvious: the ordering. This option too is seamlessly integrated in the software. The option *Order -> Order current front panel* (or: *Start ordering program*) starts an interface where all the relevant details have to be entered, after which the order is completed online.

(110238-I)

Internet Links

[1] **Front Panel designer software**
   Europe: www.schaeffer-ag.de/en/download/
   front-panel-designer.html
   USA: www.frontpanelexpress.com/download/
   front-panel-designer/index.html

[2] **Guitar Input for Multi-Effects Unit:** www.elektor.com/100923

[3] **Digital Multi-Effects Unit:** www.elektor.com/090835

[4] **Front panel download:** www.elektor.com/110238

Try it out?

While evaluating this software we designed (part of) the front panel. So this is not quite finished yet. The holes for the potentiometers for the preamp still have to be placed, just as the openings for the two 6.3-mm jack chassis sockets. We invite you to try it for yourself! You can download the front panel 'in the making' from our website [4].
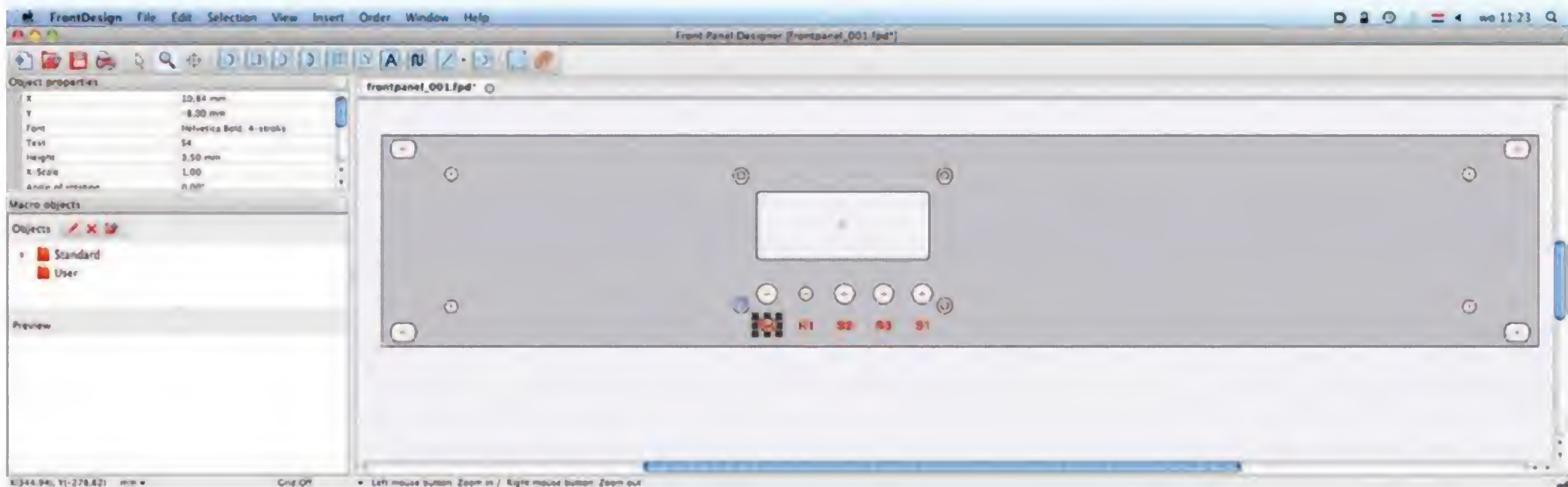
Virtual PCs

What is virtualisation and what are its uses?

Using virtualisation, more than one operating system can run at the same time on a single computer. For those of you who aren't yet familiar with this concept, you'll be surprised by the large number of applications this lends itself to. You'll also find out that free software is available on the Internet that lets you use virtualisation on your home computer.

By Evelien Snel (The Netherlands)

A virtual PC doesn't refer to the laptop carried by the avatar in **Figure 1, or** the servers that can be seen in the racks behind him. In this article, a virtual PC means 'a PC **inside** a PC': with the help of some software a PC can simulate the behaviour of another machine.

Figure 1. A virtual businessman works on his virtual laptop. In the background is a number of virtual servers.

What is virtualisation?

A very simple example of a virtual machine is the calculator in Windows/Linux/Mac OS: this program simulates all properties of a pocket calculator, apart from one: We can't put it in our pocket! When a PC simulates the behaviour of another PC this way, we speak of a virtual PC. The physical PC (the hardware) is called the 'host' and the simulated PC(s) are called 'guest(s)'. There is no need to limit ourselves to a single guest: As long as the host PC is powerful enough it can simulate the behaviour of several PCs at the same

time. We aren't restricted to a single operating system either: On a Windows PC we can reliably run a Linux system, and vice versa. Virtualisation itself isn't exactly a new technology, but in the past PCs weren't powerful enough to make a guest PC run at a reasonable speed. These days that is different: The processors are now much faster and on top of that, modern processors have special operation modes that make virtualisation easier. The most important bottleneck affecting the performance is the amount of RAM memory available.

Applications

For the electronics engineer it is very useful to have the facility to create a complete development environment in a virtual environment, which can be easily copied (backed up) and returned to (years) later for debugging purposes. The development environment in the virtual PC can have access to USB ports (and any COM ports) of the host PC via the virtualisation software. The same also applies to CD, DVD and Blue-Ray drives, network connections, etc. This means that the virtual system is hardly or not restricted in its use.

When regular backups are made of the virtual system (it's usually sufficient to make copies of the few files used by the virtualisation software for the virtual system) the continuity of the design is assured. In the event of a virus infection the complete development environment can be restored in an instant and you can continue from where you left off. Making regular backups is a good habit in this case as well.

Another application is when you want to experiment with Linux, but don't want to affect the Windows installation on your PC. You can experiment to your heart's content in a virtual PC on your PC. Nothing can go wrong with the host PC since the two operating systems are kept isolated from each other by the virtualisation software. You could use the same principle to investigate computer viruses in a virtual PC without infecting the host. In the same way, you could set up a Windows development environment on a Mac or

Figure 2. Installation of the Vmware Player.

Figure 3. The VMware Player has been installed.
Now we need a virtual machine.

Linux PC. Or any other combination of operating systems of course. It is also possible to save costs this way: Many PCs in an office environment spend most of the day waiting for some user input. It is therefore advantageous to simulate a large number of PCs in one central server, which makes much better use of its power. This saves on maintenance and energy costs. Thanks to the reduced energy consumption there may also be environmental subsidies available.

Available packages

A summary of the best-known virtualisation programs for the PC follows:

Windows virtualisation

Microsoft offers many products for virtualisation, but isn't exactly helpful when it comes to supporting operating systems other than Windows. This is not to say that it is impossible to run Linux under Microsoft Virtual PC, for example. It's just that it's not recommended.

Xensource

XenSource has recently been taken over by Citrix, which is probably the start of the commercialisation of this great product. The Xen hypervisor is an open source product developed by a group from the University of Cambridge. Xen is aimed mainly at the professional market. Despite this, a free version has always been available.

Parallels

Parallels offers a wide range of virtualisation products. There are a number of different products for use at home or work, running under either Windows or Mac or Linux. Unfortunately, there is no completely free version available. It is possible to download a trial version that works with a temporary registration key, however.

VMware

VMware, which is probably the best-known company, also offers a wide range of products. Some of those are full-featured virtualisation packages that are free to use for as long as you like.

If you want to get your teeth into virtualisation without having to get out your wallet then VMware is the perfect choice. Let's have a closer look at what they have to offer.

Free software from VMware

As with the other suppliers of virtualisation products, you can find a large number of programs on Vmware's website. The following products can be downloaded at no charge and have no time limit:

VMware Player

Until recently, the VMware Player could only run existing virtual machines, but since version 3.0 of Player it can also be used to create new virtual machines [2].

Virtual appliances

The 'Virtual appliances' from VMware are ready-made virtual machines that can be loaded in VMware Player. There is a choice from many hundreds of applications, although many have to be paid for. The Linux installations are all free though. A (free) registration on their website is required before you can download any files [3].

VMware Converter

VMware Converter can make an image of an existing PC, which can then be used as a virtual machine. It's even possible to use the source PC itself to create the image, or you could create the hard-drive image using a backup program such as Ghost, or some Linux tools.

VMware Tools

A virtual PC is completely isolated from the host system, which you find out soon enough if you try send something from the virtual PC to the host. That's just not possible! The VMware Tools are installed on the guest PC and they create a small gap in the isolation. With these tools it becomes possible to transfer files directly from the virtual system to the host system. A bit less straightforward, but still possible, is to use the network to transfer files from the virtual PC to the host PC, since both PCs are visible on the network.
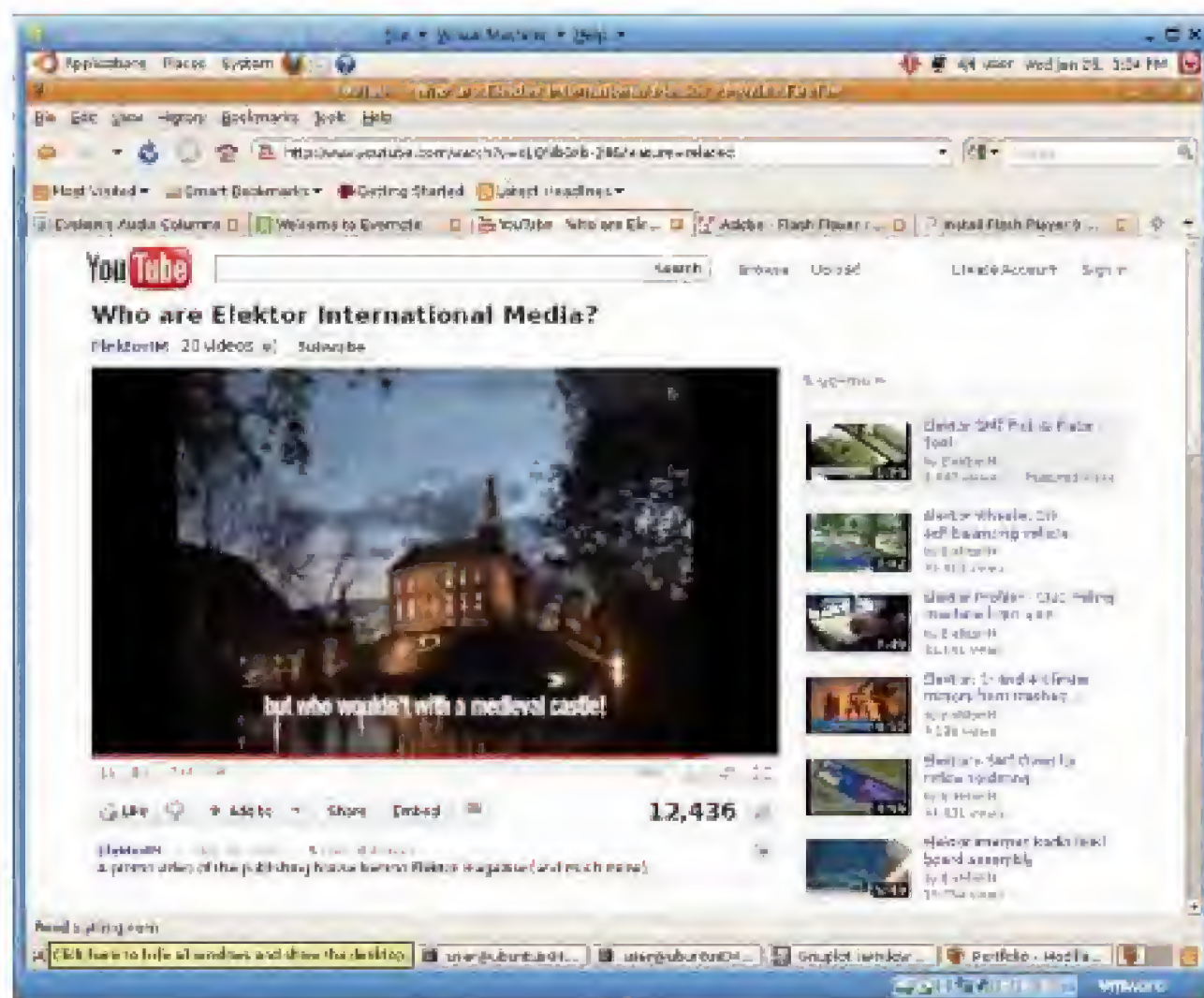
Figure 4. Watching the Elektor channel on YouTube…
On a virtual Ubuntu PC!

Figure 5. Ebony & Ivory;
Windows and Linux together in harmony on the desktop!

Get cracking

To start with, you have to download the VMware Player from [2]. This program can be installed under Windows just like any other program (**Figure 2**). Once the installation has completed we can run the program (**Figure 3**), when you'll find out that there are no virtual machines available! From the menu *File -> Download a Virtual Appliance* we can navigate straight to Vmware's website, where a number of applications can be found [3]. The button 'Operating Systems' takes you to a large number of Linux distributions. These are ready-made for use. (With each distribution mention is made of the details you need to log in to the downloaded system later. Please make a note of these!)

A good choice is 'Ubuntu 10.10 Desktop with VMware Tools' [4]. This has been made available by Chrysaor.info and it is their website that we land on when we click on 'Download'. The file has to be downloaded using BitTorrent. For those of you who have no experience of this, you can find a reputable torrent-client at [5]. (When you're installing the client, don't forget to untick those check-boxes that download toolbars and such like.)

The downloaded image has been compressed as a tgz file. In Linux this can easily be decompressed, but we don't yet have Linux! We're still busy installing it… Fortunately, it can also be decompressed using the free software from 7-zip [6].

When the file has been decompressed, we end up with a folder called 'ubuntu1010desktop.vmwarevm', which could be stored in 'My Documents\Virtual Machines' (This is the default location where VMware Player looks for them). The virtual machine can then be opened via the button 'Open Virtual Machine' and then started using 'Play Virtual Machine'. VMware Player now sees the machine for the first time and will ask if it has been moved or copied. The correct answer in this case is "I copied it". It is possible that a dialogue box pops up with information about connected USB devices; for the time being, this can be ignored and closed.
We can now log in to our new Linux system, using the login details that we noted down earlier. Open a terminal window via the menu:
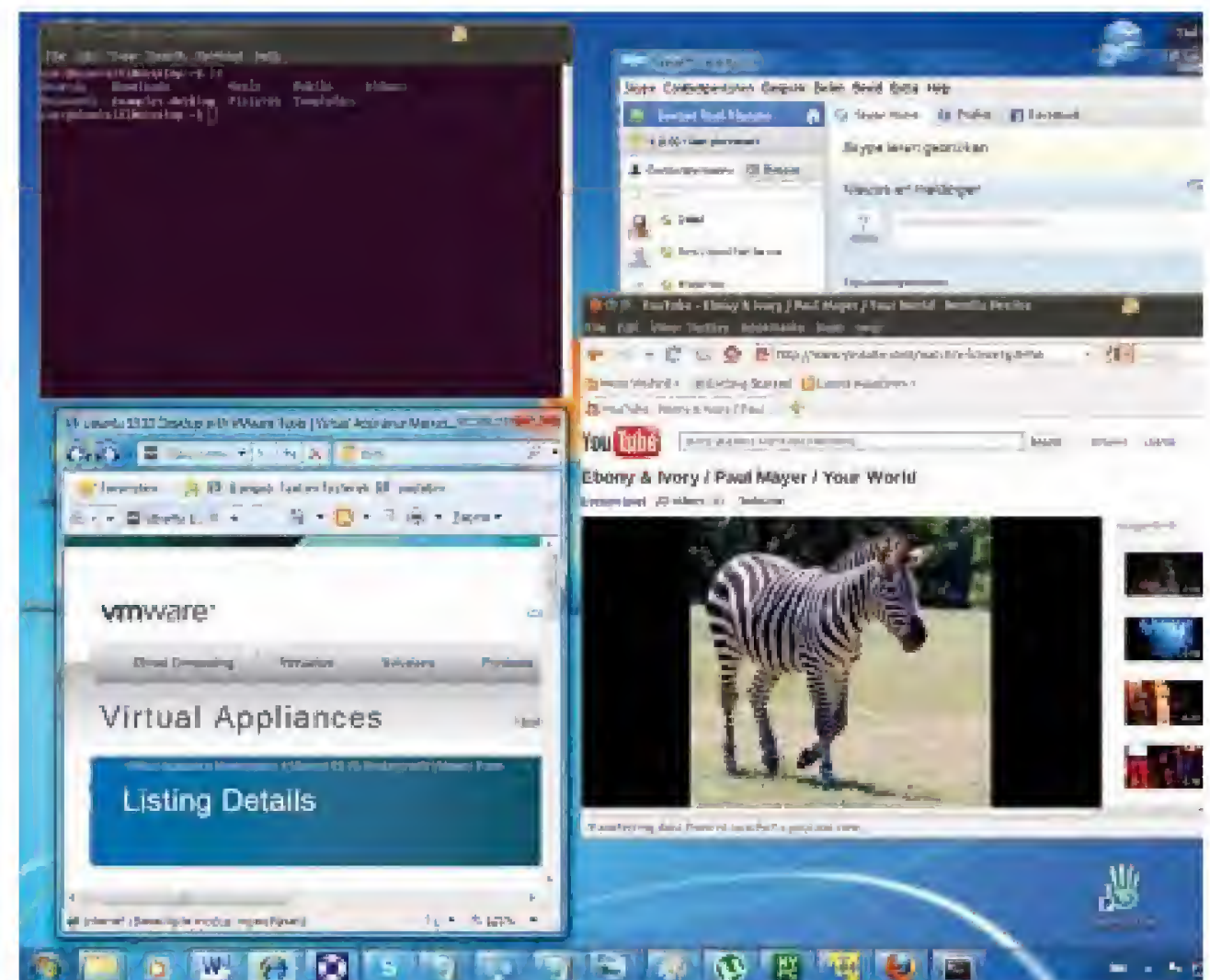
Applications -> Accessories -> Terminal. Then type in the following to start vmware-toolbox and vmware-user to enable the Copy&Paste function to and from Windows:

```
user@ubuntu1010desktop:~$ vmware-toolbox & [1]
19318
user@ubuntu1010desktop:~$ vmware-user
```

Start the virtual machine again using 'Restart…' from the menu at the top-right. The Copy&Paste function will in future be available immediately after starting up. Welcome to Linux!
Once all that has been done you can see how fast (or slow) everything works. A simple test is the viewing of YouTube movies on our virtual Linux PC (**Figure 4**).

And finally: Unity

The icing on the cake of the integration between Linux and Windows can be found via the menu in *Virtual Machine -> Enter Unity*. This detaches the windows from the Linux desktop and makes it possible for them to be shown on the standard Windows desktop (**Figure 5**). Who would have thought this possible? Windows and Linux together in harmony…

Next stop: the virtual development environment!

(100585)

Internet Links

[1]   en.wikipedia.org/wiki/Virtualization

[2]   www.vmware.com/products/player

[3]   www.vmware.com/appliances

[4]   www.vmware.com/appliances/directory/767933

[5]   www.utorrent.com

[6]   www.7zip.com

Hexadoku
Puzzle with an electronics touch

Fancy taking your mind off electronics for a spell and do a relaxing mind exercise? Then this month's Hexadoku is just the ticket to keep you busy and amused for a couple of hours. Sit up, focus the brain and enter the right numbers in the puzzle. Next, send the ones in the grey boxes to us and you automatically enter the prize draw for one of four Elektor Shop vouchers. Have fun!

The instructions for this puzzle are straightforward. Fully geared to electronics fans and programmers, the Hexadoku puzzle employs the hexadecimal range 0 through F. In the diagram composed of 16 × 16 boxes, enter numbers such that **all** hexadecimal numbers 0 through F (that's 0-9 and A-F) occur once only in each row, once in each column and in each of the 4×4 boxes (marked by the thicker black lines). A number of clues are given in the puzzle and these determine the start situation. Correct entries received enter a draw for a main prize and three lesser prizes. All you need to do is send us the numbers in the grey boxes.

Solve Hexadoku and win!

Correct solutions received from the entire Elektor readership automatically enter a prize draw for one Elektor Shop voucher worth £ 80.00 and three Elektor Shop Vouchers worth £ 40.00 each, which should encourage all Elektor readers to participate.

Participate!

Before June 1, 2011, send your solution (the numbers in the grey boxes) by email, fax or post to
Elektor Hexadoku – 1000, Great West Road – Brentford TW8 9HH United Kingdom.
Fax (+44) 208 2614447   Email: hexadoku@elektor.com

Prize winners

The solution of the March 2011 Hexadoku is: 9302F.
The PSoC 5 FirstTouch Starter Kits kindly sponsored by Cypress have been awarded to Edgar Wolff (Germany), A. van Maris (The Netherlands), Jean-Pierre Demangeon (France), Benjamin F. Creech (USA), Lars-Göran Göransson (Sweden), Alfonso Carillo Morales (Spain), Simon Eichinger (Austria), Ron Ware (UK), Alex Lo Furno (Italy) and Larry Burns (Canada).
Congratulations everyone!

Puzzle grid:

		1	7			C	6					A			5
3					7					A	F		9	E	
	E			3			8		F						
8			F	5		6	C		3		4			D	
	3	5	C			E	8	4	7			9	A	F	
	9			8		0		B	A		3			C	
	7		A	2		3		9			E			4	
	B		D			9	3		C			6			
	C		A		2	9			7			6			
	8		2		E		A		0	6		F			
	F		A		0	6		D		8			5		
E	6	0			7	9	F	2			A	3	8		
	D		3		E		A	4		1	C			9	
			4		3				D			2			
1	E		0	C			7					A			
2		8			7	B			C	1					

Solution grid:

9	D	A	2	B	F	1	6	5	C	E	0	7	4	8	3	10-2009
6	0	1	3	8	E	4	5	7	9	B	D	F	C	A	2	06-2009
8	5	7	C	9	3	0	2	F	4	6	A	1	B	E	D	05-2009
E	F	B	4	7	A	D	C	8	3	2	1	0	9	6	5	09-2009
A	7	2	0	1	8	B	E	3	6	9	5	C	D	4	F	02-2010
F	1	4	8	2	D	6	9	B	0	7	C	5	E	3	A	12-2009
B	E	C	D	A	5	F	3	2	1	4	8	6	7	9	0	11-2009
5	9	3	6	0	C	7	4	E	A	D	F	B	1	2	8	04-2010
7	C	D	E	F	6	8	B	A	2	5	4	3	0	1	9	04-2009
1	2	8	A	3	0	9	7	D	F	C	E	4	5	B	6	02-2009
4	3	9	5	C	1	2	D	6	7	0	B	8	A	F	E	01-2009
0	B	6	F	5	4	E	A	9	8	1	3	D	2	7	C	03-2009
C	8	5	1	E	7	A	F	4	D	3	2	9	6	0	B	03-2010
2	6	F	B	4	9	5	0	C	E	8	7	A	3	D	1	01-2010
3	4	0	9	D	2	C	8	1	B	A	6	E	F	5	7	05-2010
D	A	E	7	6	B	3	1	0	5	F	9	2	8	C	4	06-2010

Tektronix 564 Storage Oscilloscope (1963)

By Jan Buiting (Elektor UK/US Editorial)

It's doubtful you can consider yourself an old hand at electronics if do not instantly recognise a Tektronix 500 series oscilloscope. These beasts in their light blue cases were all over electronics labs, colleges and other 'places scientific' for more than two decades before being scrapped or traded in for gear with transistors inside. Many users still resent that. On one website dedicated to Tektronix

tube-based test equipment the word 'transistor' is banned, with a single occurrence only of "trnsstr".

It's easy to see why Tek 500 series 'scopes now stir up nostalgic feelings of 1960s and 1970s 'college & lab' days. For one thing, they are incredibly solid builds compared to the current generation products with their plastic covers, rubber knobs and arcane texts on the screen. Add to that the massive weight (the 564 without plug-ins weighs in at 13.8 kgs / 30.5 lbs), the noise of the extractor fan (not on all models), the razor sharp image on the CRT screen, the famous plug-ins and the clic-clac feel and sound of the immense black and red controls on the front panel— and you have a non-erasable item in your memory.

In the 1970s and 1980s my Tek 'scope was a Type 535 (I think) with a two-channel channel plug-in. Compared to the DIY grot my friends were using, my 535's frequency range was such it got hauled around the province in my dad's car, 'Scope-Mobile' trolley and all, to do complex measurements and observe elusive glitches mostly in Elektor projects with CMOS ICs. I also used it to heat my little workshop in the attic. One channel I never managed to get stable vertically though on the vertical plug-in despite exchanging tubes. It was the first time I saw nuvistors and gold plated contacts.

Speaking of contacts, most parts and wires in the Tek 500 series oscilloscopes are soldered using 3% silver bearing solder to guarantee a joint that would last a lifetime. One of the HV supply panels inside the 'scope has a plastic spool with spare silver-bearing solder secured to it for repair purposes. At the age of 20, having written for a new supply of that precious solder (in the kindest US American words I could manage), after about a month I received a package with two spools inside from Tektronix' European plant on Guernsey, Channel Islands. There was also a cover letter with courteous advice on soldering as well as two 'revision notes'. My mother said it was a miracle and "so like the Tommies in 1945 — always good humoured"; to which I replied "I reckon it's just service-friendly".

According to the instrument manuals, plug-in units can be changed without switching the main instrument off, so 'hot-plugging' is not a 1990s invention. I never had the guts to try this out for fear of
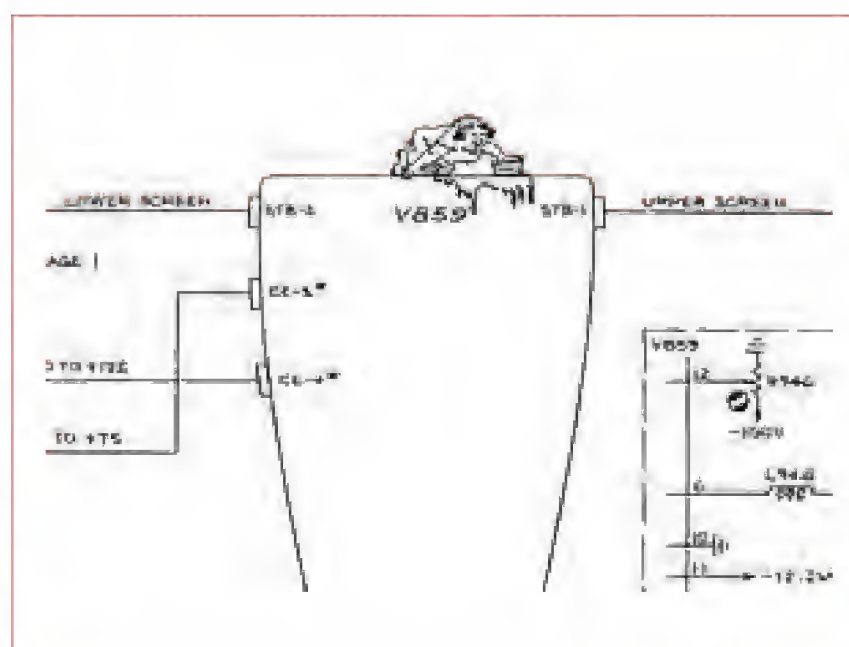
damaging my precious 535 and having to borrow another 'scope to repair it. After about 10 years I sold it on — with a profit.

The 564 shown here is a lightweight and a dwarf compared to the 535. I got it from my neighbour in exchange for a bucket of prunes last autumn. The instrument was excavated from a pile of assorted household goods and furniture in a cellar in her parents' apartment building. For plug-ins it contained the 2B67 timebase and the 3A6 dual-trace amplifier. Having done the usual cleaning, dusting, vacuum cleaning and careful initial powering up of the 564 I decided to do a round of 'Tektronix vintage' searching on the Internet. To my amazement I came across a method of thoroughly cleaning an oscilloscope actually applied by Tektronix themselves at Factory Service Center and offered as a service to clients. Remarkably, instruments got washed using warm distilled water and a lab glassware cleaning detergent, with critical parts sealed in plastic. The method was maintained and publicized till the late 1970s for the 7000-series with trnsstrs and even ICs [1]. The crux is not the washing proper but the controlled drying of the instrument in an oven for 24 hours at a low temperature. Tek dryly write: *Place the instrument (with washed plug-ins installed) in the oven and dry for at least 24 hours (Oven makes good storage place until item is needed to be worked on. More drying is o.k.).* The washing and drying is followed by a round of lubrication and greasing of switches, pots and motors using Tektronix Lubrication kit p/n 003-0342-01 (what else!). I decided to wash my car instead of the 564.

Opening the case of a tubed Tektronix oscilloscope is like opening the bonnet of a 1930s American car with a big engine in the nose. Each of the two partly perforated covers at the sides of the instrument can be removed after loosening just two screws. You are greeted by a dozen or so tubes, dust, PATENTs printed on the chassis and the smell of vintage electronics 'US style'. In some cases, the plug-ins contain more tubes and trnsstrs than the main chassis. My 564 turned out to work okay on one channel (trace). This is a storage oscilloscope with zero (no) RAM. Instead, storage screens, flood guns and collimation electrodes inside the purpose-designed CRT provide a primitive display memory. Once you see the signal you want to 'freeze' on the CRT, you use the ERASE, INTENSITY and STORE controls in a certain sequence. A not too sharp image

appears and is retained in phosphor (Tek claim) for at least one hour. My 564 managed about 15 minutes before the (poor) image had vanished completely. Some recalibration may be in order and failing that I'm planning on writing to Oregon for a replacement CRT and some spare solder tin. The stored image is erased by moving the lever to ERASE whereupon the screen flashes a bright green. The stored trace may be used for good/fault comparison to the live trace as well as for photography. The latter application is discussed in detail in the 564 Instruction Manual.

In good US tradition, retired Tektronix staff have nice anecdotes to share like: mail was received at the factory from users in what was politely described as "technologically less advanced parts of the world". Although the envelopes were addressed: *Tektronix Inc., Portland, Oregon, USA, VERTICAL*, they got duly delivered by US Mail. The Instruction Manual that came with the 564 is a typical 1960s US electronics engineering product. Extensive, comprehensive and nicely illustrated, with full descriptions of the electrical operation of the instrument, maintenance and repair guidance. Frankly I wasn't expecting to find any light-hearted stuff in the manual but on the very last page the *564 Washer Woman* suddenly appears cleaning the face of CRT V859. Engineers' humour. On ClassicTek.org other funny characters can be seen like the *7A26 Cowboy*, the *184 Disabled Man* and the *502 Mountain Climber*. Gotcha — my mother was right after all.

(100920)

Internet References

[1]   www.classictek.org

[2]   www.r-type.org/exhib/aaa0273.htm

For further reading

The Museum of Tek Scopes:
    www.chiark.greenend.org.uk/scopes/tek.html

BarryTech:
    www.barrytech.com/tektronix/vintage/tekvintage.html

Tek Museum:
    http://tekmuseum.ebaman.com/main.html#tline

Retronics is a monthly column covering vintage electronics including legendary Elektor designs. Contributions, suggestions and requests are welcomed; please send an email to editor@elektor.com
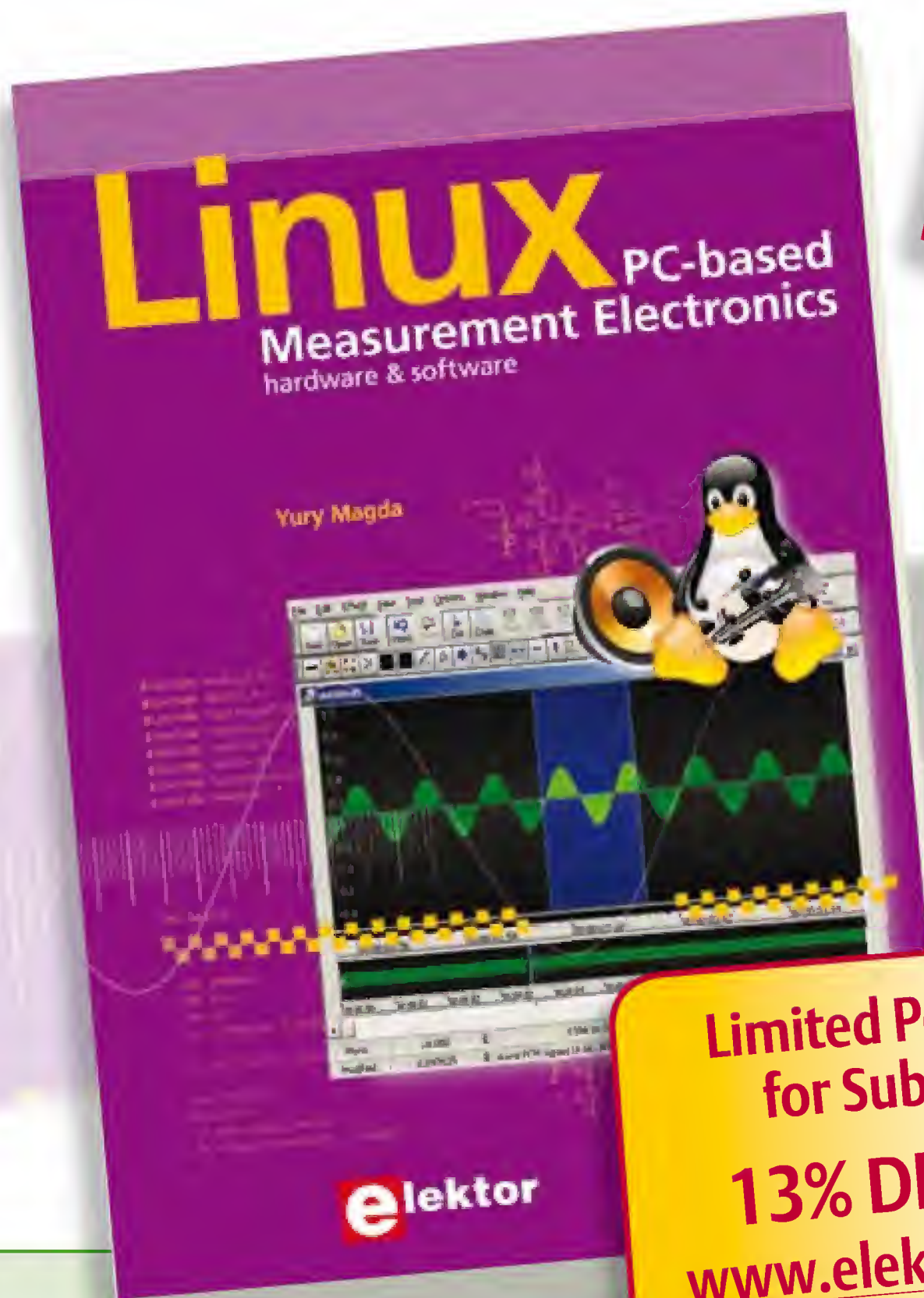
products and services directory

Going Strong

A world of electronics from a single shop!

NEW!

Limited Period Offer for Subscribers!

13% DISCOUNT
**www.elektor.com/may**

Books

Improve your skills in electronics, programming, networking and wireless design

Linux – PC -based Measurement Electronics

This book is intended as a highly-practical guide for people wishing to build measurement and control systems to be controlled by a local or remote Personal Computer running the Linux operating system. Both hardware and software aspects of designing typical embedded systems are covered in detail with schematics, code listings and full descriptions. There is no need to buy special equipment or expensive software tools in order to create embedded projects covered in this book. You can build such quality systems quickly using popular low-cost electronic components and free distributed or low-cost software tools. Many projects in this book are developed using Matlab® being a very popular worldwide computational tool for research in engineering and science. The book provides a detailed description of how to combine the power of Matlab® with practical electronics. With an emphasis on learning by doing, readers are encouraged by examples to program with ease; the book provides clear guidelines as to the appropriate programming techniques "on the fly". Complete and well-documented source code is provided for all projects.

Approx. 250 pages • ISBN 978-1-907920-03-5 • £29.50 • US $47.60
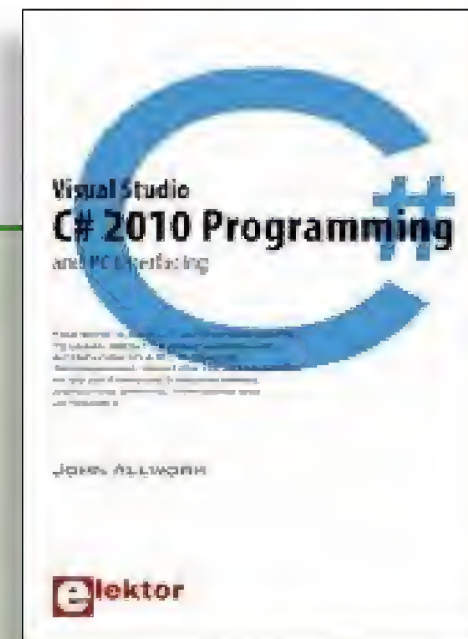
Visual Studio

C# 2010 Programming and PC interfacing

This book is aimed at anyone who wants to learn about C# programming and interfacing to a PC. It covers programming concepts from the basics to object oriented programming, displaying graphs, threading and databases. The book is complete with many full program examples, self assessment exercises and links to supporting videos. All code examples used are available – free of charge – from a special support website.

306 pages • ISBN 978-0-905705-95-8
£29.50 • US $47.60
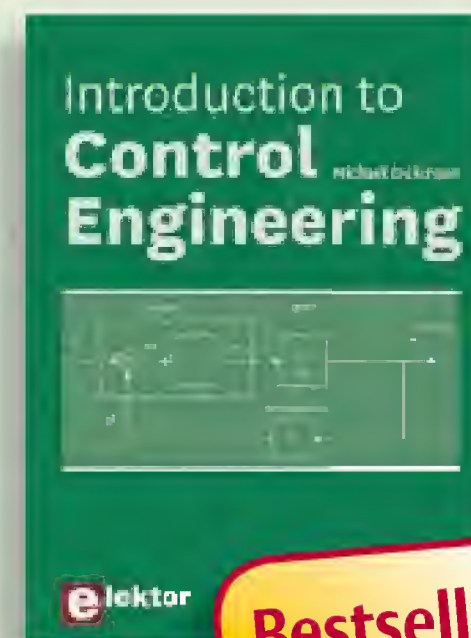
Bestseller!

Solutions for control system applications

Introduction to Control Engineering

This book is intended as a source of reference for hardware and software associated with instrumentation and control engineering. Examples are presented from a range of industries and applications. Throughout the book, circuit diagrams and software listings are described, typical of many measurement and control applications. The hardware and software designs may be used as a basis for application by the reader. The book contains examples of PIC, PLC, PAC and PC programming.

164 pages • ISBN 978-0-905705-99-6
£27.50 • US $44.40
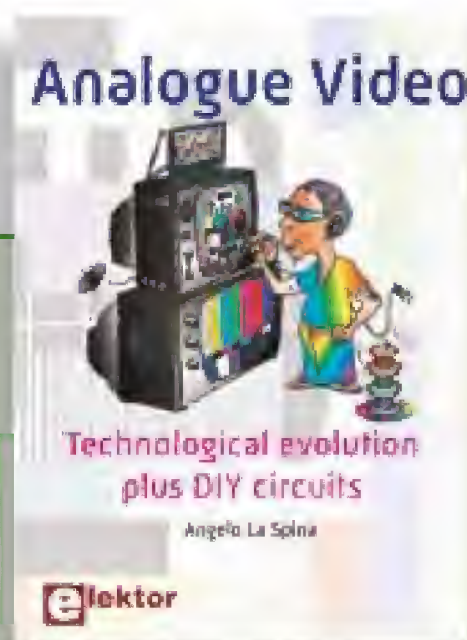
Analogue Video

Technological evolution plus DIY circuits

Analogue Video

This book is intended for electronics enthusiasts and professionals alike, who want a much deeper understanding of the incredible technology conquests over the pre-digital decades that created video. It details evolution of analogue video electronics and technology from the first electro-mechanical television, through advancements in Cathode Ray Tubes, transistor circuits and signal processing, up to the latest analogue, colour-rich TV, entertainment devices and calibration equipment.

222 pages • ISBN 978-0-905705-96-5
£26.50 • US $42.80

Associated starter kit available

ARM Microcontrollers

This is the perfect book for people who want to learn C and who want to use an mbed ARM microcontroller in an easy and fun way. The mbed NXP LPC1768 uses cloud technology. This means you do not need to install software on your PC in order to program the mbed! The only thing you need is a browser and a USB port on your PC. No previous experience or knowledge required. You can get access to your project from any PC anywhere in the world and continue working on it. When you are done a few mouse clicks transfer the program to your mbed hardware.

250 pages • ISBN 978-0-905705-94-1
£29.50 • US $47.60

Enhanced second edition: 180 new pages

Design your own Embedded Linux Control Centre on a PC

The main system described in this book reuses an old PC, a wireless mains outlet with three switches and one controller, and a USB webcam. All this is linked together by Linux. This book will serve up the basics of setting up a Linux environment – including a software development environment – so it can be used as a control centre. The book will also guide you through the necessary setup and configuration of a webserver, which will be the interface to your very own home control centre. New edition enhancements include details of extending the capabilities of your control center with ports for a mobile phone (for SMS messaging) and the Elektor "thermo snake" for low-cost networked real-time thermal monitoring of your house and outbuildings. Now you can additionally also send all kinds of useful temperature and sensor warnings to a mobile phone. All software needed will be available at the Elektor website.

416 pages • ISBN 978-1-907920-02-8
£34.50 • US $55.70

More information on the Elektor Website:

www.elektor.com

Elektor
Regus Brentford
1000 Great West Road
Brentford
TW8 9HH
United Kingdom
Tel.: +44 20 8261 4509
Fax: +44 20 8261 4447
Email: sales@elektor.com

More than 70,000 components

CD Elektor's Components Database 6

This CD-ROM gives you easy access to design data for over 7,800 ICs, more than 35,600 transistors, FETs, thyristors and triacs, just under 25,000 diodes and 1,800 optocouplers. The program package consists of eight databanks covering ICs, transistors, diodes and optocouplers. A further eleven applications cover the calculation of, for example, zener diode series resistors, voltage regulators, voltage dividers and AMV's. A colour band decoder is included for determining resistor and inductor values. All databank applications are fully interactive, allowing the user to add, edit and complete component data.

ISBN 978-90-5381-258-7
£24.90 • US $40.20

More than 25 projects based on the Elektor ATM18 board

CD ATM18 Collection

This CD-ROM contains all articles from the popular ATM18-CC2 series published in Elektor magazine. From RFID Reader and Bluetooth linking right up to a chess computer! Project software and PCB layouts in PDF format are included. What's more, this CD also contains a Bascom AVR programming course and helpful supplementary documentation.

ISBN 978-0-905705-92-7
£24.50 • US $39.60

CD/DVD-ROMS

RFID, NFC, Zigbee, GPS and more

DVD Wireless Toolbox

On this DVD-ROM you'll find a number of documents and tools that will enable you to add wireless data exchange to your electronics systems. In accordance with the principle of our Toolbox series, we've brought together technical documentation (spec. sheets, application notes, user guides, etc.) on various devices according to the frequency and/or protocol used. All of the documents are PDF files (in English). Browsing around the DVD is made easy by an HTML menu. Finally, this Wireless Toolbox DVD contains a collection of articles on this topic (RFID, xBee, DCF77, GPS, infrared, etc.) that have appeared in Elektor magazine.

ISBN 978-90-5381-268-6
£28.50 • US $46.00

Bestseller!

All articles in Elektor Volume 2010

DVD Elektor 2010

This DVD-ROM contains all editorial articles published in Volume 2010 of the English, Spanish, Dutch, French and German editions of Elektor. Using the supplied Adobe Reader program, articles are presented in the same layout as originally found in the magazine. An extensive search machine is available to locate keywords in any article. With this DVD you can also produce hard copy of PCB layouts at printer resolution, adapt PCB layouts using your favourite graphics program, zoom in / out on selected PCB areas and export circuit diagrams and illustrations to other programs.

ISBN 978-90-5381-267-9
£23.50 • US $37.90
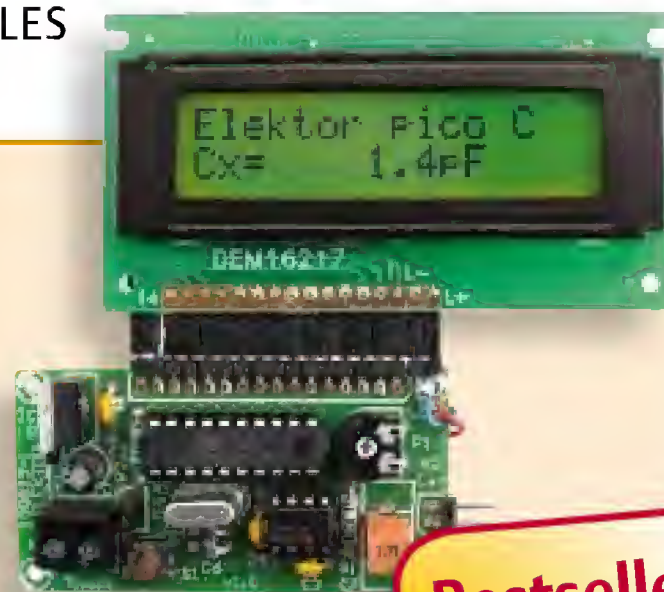
Bestseller!

Pico C Meter

(April 2011)

RF and radio repair fans probably do need to be told, but when it comes to measurements below 200 pF or so, modern DMMs will produce coarse if not ridiculous results. Elektor's purpose-designed Pico C does a far better job. Beating many DMMs hands down, this little instrument easily and accurately measures capacitances down to fractions of a picofarad!

Kit of parts incl. Elektor Project Case, programmed microcontroller, LCD and PCB

Art.# 100823-71 • £73.40 • US $118.40

Wireless OBD-II

(April 2011)

The cheapest way to diagnose faults on a modern car is to connect its OBD-II interface to a (notebook) PC running suitable diagnostics software. However, a wired connection is not always the most suitable, and selfcontained OBD testers are a rather expensive and less flexible alternative to using a PC. An interesting option is a wireless OBD interface with a radio interface to a PC: this homebrew solution allows the choice of using either Bluetooth or ZigBee.

OBD2- Zigbee or Bluetooth interface kit with all parts and enclosure

Art.# 100872-71
£125.00 • US $201.70 (Zigbee)
Art.# 100872-72
£125.00 • US $201.70 (Bluetooth)

SatFinder

(March 2011)

Those of you who regularly need to realign a satellite TV dish will find this gadget extremely valuable. Caravan owners and campers on long journeys who crave their home TV channels can now keep up with developments in sports, news and the soaps back home with the help of the SatFinder. This GPS based design includes a database containing positional information of a number of popular TV satellites. With the help of GPS data it calculates the precise angles to find the satellite first time!

Kit of parts including Controller, display and PCB (European Version)

Art.# 100699-71 • £71.20 • US $114.90
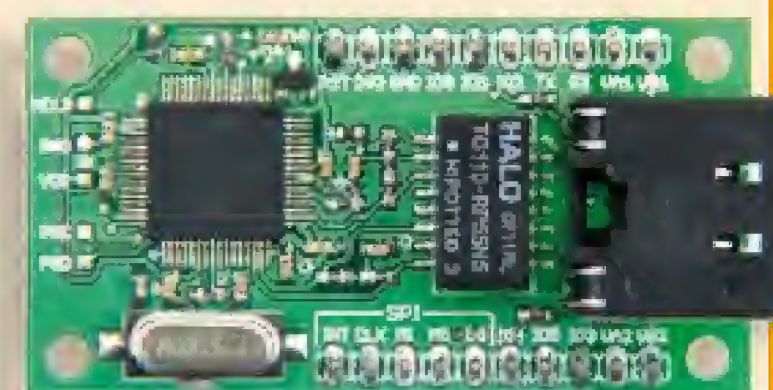
NetWorker

(December 2010)

An Internet connection would be a valuable addition to many projects, but often designers are put off by the complexities involved. The 'NetWorker', which consists of a small printed circuit board, a free software library and a ready-to-use microcontroller-based web server, solves these problems and allows beginners to add Internet connectivity to their projects. More experienced users will benefit from features such as SPI communications, power over Ethernet (PoE) and more.

Module, ready assembled and tested

Art.# 100552-91 • £53.00 • US $85.50

LED Bicycle Headlight

As many a cyclist will be able to confirm, the average amount of light produced by a bicycle headlight is disappointing. The Readers Project presented in the June 2011 edition is enlightening in more than one way. The headlight to be described has four Cree LEDs which together assure a bright beam worth 600 lumens. Power is provided by four Li-ion batteries of 2200 mAh each. A microcontroller looks after the LED controlling and charging of the batteries.
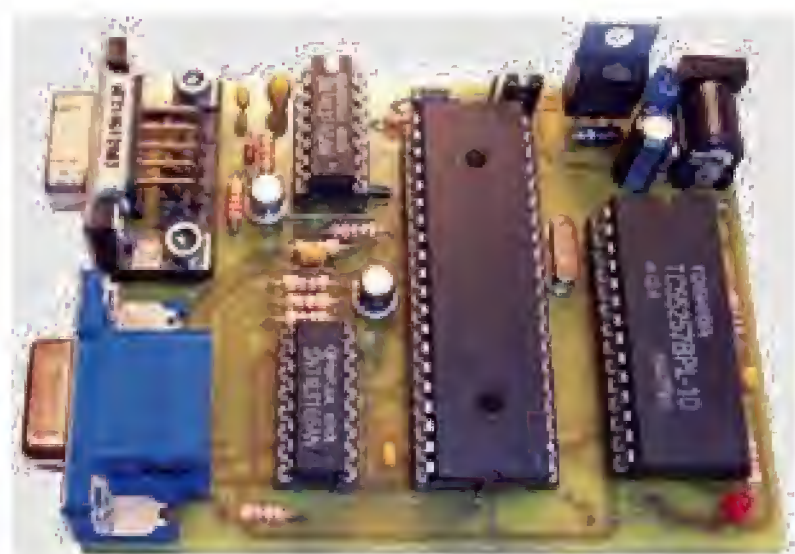
VGA Add-On for Microcontrollers

Many projects require a large amount of information to be displayed, but the size of the display itself is often a problem. One solution is to use an old 14" or 15" computer monitor that's been scrapped but is still working. The VGA board described in the June 2011 edition lets you do just this, and is compatible with any microcontroller that has a serial port.

Android on a Beagle Board

Although the free open-source Android operating system is actually designed for mobile phones, it may also very well work on other systems, such as a TI Beagle board with its low-power ARM processor. Next month we will tell you more about Android, its origins and the methods of developing custom applications. Furthermore, we show how to create a complete Android system yourself. (Photo: BeagleBoard.org)

Article titles and magazine contents subject to change; please check the Magazine tab on www.elektor.com

Elektor UK/European April 2011 edition: on sale May 18 2011.   *Elektor USA April 2011 edition: published May 15, 2011.*

w.elektor.com www.elektor.com www.elektor.com www.elektor.com www.elektor.com wv

Elektor on the web

All magazine articles back to volume 2000 are available online in pdf format. The article summary and parts list (if applicable) can be instantly viewed to help you positively identify an article. Article related items are also shown, including software downloads, circuit boards, programmed ICs and corrections and updates if applicable. Complete magazine issues may also be downloaded.

In the Elektor Shop you'll find all other products sold by the publishers, like CD-ROMs, DVDs, kits, modules, equipment, tools and books. A powerful search function allows you to search for items and references across the entire website.

Also on the Elektor website:

- Electronics news and Elektor announcements
- Readers Forum
- PCB, software and e-magazine downloads
- Time limited offers
- FAQ, Author Guidelines and Contact

Description	Price each	Qty.	Total	Order Code
Linux – PC-based Measurement Electronics **NEW**	£29.50			
Assembly Language Essentials **NEW**	£29.50			
Design your own Embedded Linux Control Centre on a PC **NEW**	£34.50			
CD ATM18 Collection **NEW**	£29.50			
CD Elektor's Components Database 6 **NEW**	£24.90			
Analogue Video	£26.50			
Introduction to Control Engineering	£27.50			
DVD Elektor 2010	£23.50			
DVD Wireless Toolbox	£28.50			
	Sub-total			

Prices and item descriptions subject to change. The publishers reserve the right to change prices without prior notification. Prices and item descriptions shown here supersede those in previous issues. E. & O.E.

P&P

Total paid

Name

Address + Post code

Tel.   Email

Date   –   –   Signature

EL05

METHOD OF PAYMENT
(see reverse before ticking as appropriate)

☐ Bank transfer

☐ Cheque
(UK-resident customers ONLY)

☐ VISA   ☐ MasterCard EUROCARD

Expiry date: _____

Verification code: _____ .

Please send this order form to*
(see reverse for conditions)

Elektor

Regus Brentford

1000 Great West Road

Brentford TW8 9HH

United Kingdom

Tel.: +44 20 8261 4509
Fax: +44 20 8261 4447
www.elektor.com
order@elektor.com

*USA and Canada residents should use S prices, and send the order form to:
Elektor US
PO Box 180
Vernon CT 06066
USA
Phone: 860-875-2199
Fax: 860-871-0411
E-mail: order@elektor.com

Yes, I am taking out an annual subscription to Elektor and receive the E-book Microcontroller Basics totally free!*

(Please fill in your emailaddress below)

I would like:

☐ Standard Subscription (11 issues)

☐ Subscription-Plus
(11 issues plus the Elektor Volume 2011 DVD-ROM + exclusive access to www.elektor-plus.com) **TIP**

* Offer available to Subscribers who have not held a subscription to Elektor during the last 12 months. Offer subject to availability. See reverse for rates and conditions.

Name

Address + Post code

Tel.   Email

Date   –   –   Signature

EL05

METHOD OF PAYMENT
(see reverse before ticking as appropriate)

☐ Bank transfer

☐ Cheque
(UK-resident customers ONLY)

☐ VISA   ☐ MasterCard EUROCARD

Expiry date: _____

Verification code: _____

Please send this order form to

Elektor

Regus Brentford

1000 Great West Road

Brentford TW8 9HH

United Kingdom

Tel.: +44 20 8261 4509
Fax: +44 20 8261 4447
www.elektor.com
service@elektor.com

ORDERING INSTRUCTIONS, P&P CHARGES

All orders, except for subscriptions (for which see below), must be sent BY POST or FAX to our Brentford address using the Order Form overleaf. Online ordering: www.elektor.com/shop

Readers in the USA and Canada should send orders, except for subscriptions (for which see below), to the USA address given on the order form. Please apply to Elektor US for applicable P&P charges. Please allow 4-6 weeks for delivery.

Orders placed on our Brentford office must include P&P charges (Priority or Standard) as follows: Europe: £6.00 (Standard) or £7.00 (Priority) Outside Europe: £9.00 (Standard) or £11.00 (Priority)

HOW TO PAY

All orders must be accompanied by the full payment, including postage and packing charges as stated above or advised by Customer Services staff.

Bank transfer into account no. 40270211 held by Elektor International Media BV with The Royal Bank of Scotland, London. IBAN: GB96 ABNA 4050 3040 2702 11. BIC: ABNAGB2L. Currency: sterling (UKP).
Please ensure your full name and address gets communicated to us.

Cheque sent by post, made payable to Elektor Electronics. We can only accept sterling cheques and bank drafts from UK-resident customers or subscribers. We regret that no cheques can be accepted from customers or subscribers in any other country.

GCredit card VISA and MasterCard can be processed by mail, email, web, fax and telephone. Online ordering through our website is SSL-protected for your security.

COMPONENTS

Components for projects appearing in Elektor are usually available from certain advertisers in this magazine. If difficulties in the supply of components are envisaged, a source will normally be advised in the article. Note, however, that the source(s) given is (are) not exclusive.

TERMS OF BUSINESS

Delivery Although every effort will be made to dispatch your order within 2-3 weeks from receipt of your instructions, we can not guarantee this time scale for all orders. **Returns** Faulty goods or goods sent in error may be returned for replacement or refund, but not before obtaining our consent. All goods returned should be packed securely in a padded bag or box, enclosing a covering letter stating the dispatch note number. If the goods are returned because of a mistake on our part, we will refund the return postage. **Damaged goods** Claims for damaged goods must be received at our Brentford office within 10-days (UK); 14-days (Europe) or 21-days (all other countries). **Cancelled orders** All cancelled orders will be subject to a 10% handling charge with a minimum charge of £5.00. **Patents** Patent protection may exist in respect of circuits, devices, components, and so on, described in our books and magazines. Elektor does not accept responsibility or liability for failing to identify such patent or other protection. **Copyright** All drawings, photographs, articles, printed circuit boards, programmed integrated circuits, diskettes and software carriers published in our books and magazines (other than in third-party advertisements) are copyright and may not be reproduced or transmitted in any form or by any means, including photocopying and recording, in whole or in part, without the prior permission of Elektor in writing. Such written permission must also be obtained before any part of these publications is stored in a retrieval system of any nature. Notwithstanding the above, printed-circuit boards may be produced for private and personal use without prior permission. **Limitation of liability** Elektor shall not be liable in contract, tort, or otherwise, for any loss or damage suffered by the purchaser whatsoever or howsoever arising out of, or in connexion with, the supply of goods or services by Elektor other than to supply goods as described or, at the option of Elektor, to refund the purchaser any money paid in respect of the goods. **Law** Any question relating to the supply of goods and services by Elektor shall be determined in all respects by the laws of England.

January 2011

SUBSCRIPTION RATES FOR ANNUAL SUBSCRIPTION

	Standard	Plus
United Kingdom & Ireland	£51.00	£63.50
Surface Mail Rest of the World	£65.00	£77.50
Airmail Rest of the World	£82.00	£94.50
USA & Canada	See www.elektor.com/usa for special offers	

HOW TO PAY

Bank transfer into account no. 40270211 held by Elektor International Media BV with The Royal Bank of Scotland, London. IBAN: GB96 ABNA 4050 3040 2702 11. BIC: ABNAGB2L. Currency: sterling (UKP). Please ensure your full name and address gets communicated to us.

Cheque sent by post, made payable to Elektor Electronics. We can only accept sterling cheques and bank drafts from UK-resident customers or subscribers. We regret that no cheques can be accepted from customers or subscribers in any other country.

Credit card VISA and MasterCard can be processed by mail, email, web, fax and telephone. Online ordering through our website is SSL-protected for your security.

SUBSCRIPTION CONDITIONS

The standard subscription order period is twelve months. If a permanent change of address during the subscription period means that copies have to be despatched by a more expensive service, no extra charge will be made. Conversely, no refund will be made, nor expiry date extended, if a change of address allows the use of a cheaper service.

Student applications, which qualify for a 20% (twenty per cent) reduction in current rates, must be supported by evidence of studentship signed by the head of the college, school or university faculty.
A standard Student Subscription costs £40.80, a Student Subscription-Plus costs £53.30 (UK only).

Please note that new subscriptions take about four weeks from receipt of order to become effective.
Cancelled subscriptions will be subject to a charge of 25% (twenty-five per cent) of the full subscription price or £7.50, whichever is the higher, plus the cost of any issues already dispatched. Subscriptions cannot be cancelled after they have run for six months or more.

INDEX OF ADVERTISERS